

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-319562

(43)Date of publication of application : 12.12.1997

(51)Int.Cl.

G06F 9/06

G06F 15/16

(21)Application number : 08-198773

(71)Applicant : MITSUBISHI ELECTRIC CORP

(22)Date of filing : 29.07.1996

(72)Inventor : SUZUKI HISAAKI
OGINO TADASHI
TAKAYAMA SHIGENOBU
FUJIWARA SATOKO

(30)Priority

Priority number : 07333710

Priority date : 21.12.1995

Priority country : JP

07333711

21.12.1995

08 72466

27.03.1996

JP

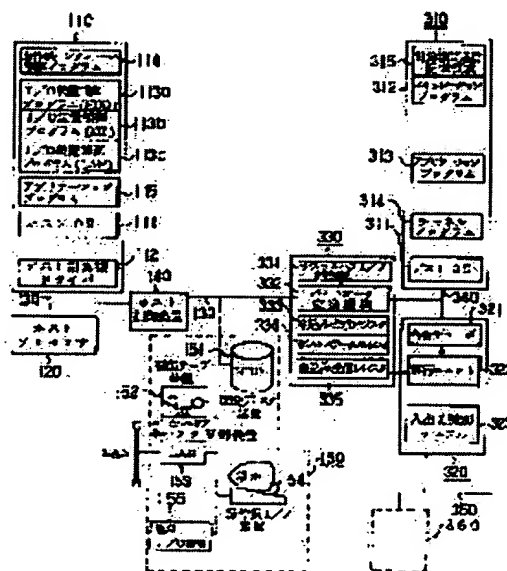
JP

(54) COMPOSITE COMPUTER SYSTEM AND METHOD FOR INSTALL/ UNINSTALL TO COMPOSITE COMPUTER SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a high-performance and low-cost composite computer system in flexible configuration.

SOLUTION: A host computer has a guest computer driver 112 for simultaneously controlling operations between guest computers, host main storage device 110 for storing an I/O device control program 113 for controlling data transfer between any specified host I/O device 150 and a guest main storage device 310 corresponding to an input/output request from the guest computer, and host controller 140 for performing data transfer control and the control of interruption to a host processor 120. The guest computer has the guest main storage device 310 including a shared inter-computer communication storage area 315 for setting an emulation program 312 having an input/output instruction executing function and interruption request information or the like and a guest controller 330 for performing data transfer control and the control of interruption to a guest processor 320, and based on an input/output identification table 323, data are inputted/outputted to the specified host I/O device 150 which can be shared.



LEGAL STATUS

(51) Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/06	4 1 0		G 0 6 F 9/06	4 1 0 B
15/16	3 7 0		15/16	3 7 0 Z

審査請求 有 請求項の数20 O L (全 34 頁)

(21) 出願番号 特願平8-198773

(22) 出願日 平成8年(1996)7月29日

(31) 優先権主張番号 特願平7-333710

(32) 優先日 平7(1995)12月21日

(33) 優先権主張国 日本 (J P)

(31) 優先権主張番号 特願平7-333711

(32) 優先日 平7(1995)12月21日

(33) 優先権主張国 日本 (J P)

(31) 優先権主張番号 特願平8-72466

(32) 優先日 平8(1996)3月27日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72) 発明者 鈴木 壽明

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

(72) 発明者 荻野 正

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

(72) 発明者 ▲たか▼山 茂伸

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

(74) 代理人 弁理士 吉田 研二 (外2名)

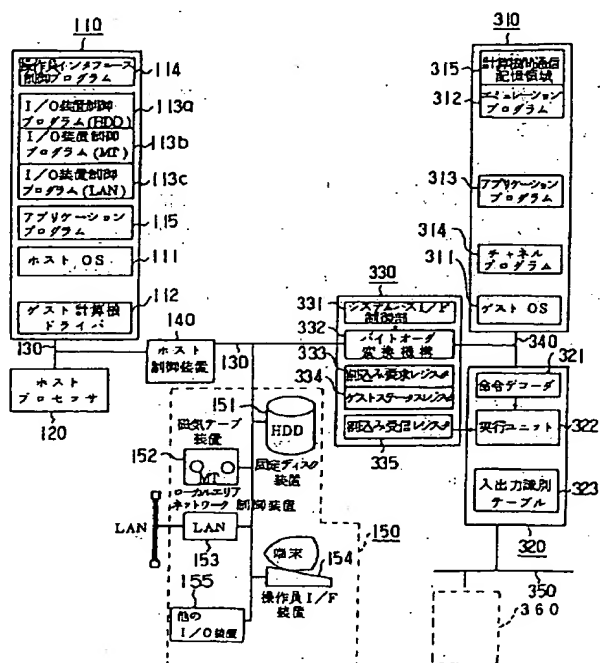
最終頁に続く

(54) 【発明の名称】 複合計算機システム及び複合計算機システムへのインストール／アンインストール方法

(57) 【要約】

【課題】 高性能、低コストかつ柔軟な構成の複合計算機システムを提供する。

【解決手段】 ホスト計算機は、ゲスト計算機間の動作を一括制御するゲスト計算機ドライバ112及びゲスト計算機からの入出力要求により特定のホストI/O装置150とゲスト主記憶装置310間のデータ転送を制御するI/O装置制御プログラム113を記憶するホスト主記憶装置110と、データ転送制御及びホストプロセッサ120に対する割込み制御を行うホスト制御装置140とを有する。ゲスト計算機は、入出力命令実行機能を持つエミュレーションプログラム312及び割込み要求情報等が設定される共有化された計算機間通信記憶領域315を含むゲスト主記憶装置310と、データ転送制御及びゲストプロセッサ320に対する割込み制御を行うゲスト制御装置330とを有し、入出力識別テーブル323に基づき特定した共用可能なホストI/O装置150に入出力を行う。



【 特許請求の範囲】

【 請求項1 】 ホスト 計算機と ゲスト 計算機とにより構成される複合計算機システムにおいて、

前記ホスト 計算機は、

プリ エンプティブ・ マルチタスク 処理機能を有するオペレーティングシステム及びアプリケーションプログラムが動作するホスト プロセッサと、

システムバスに接続された一群の入出力装置と、

前記入出力装置に対するアクセスを行うために前記入出力装置それぞれに対応して設けられ、前記ゲスト 計算機 10 間のデータ転送を行う 入出力装置制御手段と、

システムバス上のデータ転送制御及び前記ホスト プロセッサに対する割込み制御を行う ホスト 制御手段と、

前記ゲスト 計算機を前記システムバスに接続された1つの入出力装置とみなして前記ホスト 計算機における前記ゲスト 計算機の制御を一括して行うゲスト 計算機制御手段と、

前記ゲスト 計算機を接続するシステムバスと、

を有し、

前記ゲスト 計算機は、

前記ホスト プロセッサと独立したオペレーティングシステム及びアプリケーションプログラムが動作するゲスト プロセッサと、

前記ゲスト プロセッサによるエミュレーション命令を実行するエミュレーション手段と、

前記システムバスを接続し、前記ホスト 計算機との間のデータ転送制御及び前記ゲスト プロセッサに対する割込み制御を行う ゲスト 制御手段と、

前記ホスト 計算機からアクセス可能であり各種情報を相互に交換する計算機間通信記憶領域を含むゲスト 主記憶 30 手段と、

を有し、前記ゲスト プロセッサを用いることなく前記入出力装置に対してアクセスを行うことを特徴とする複合計算機システム。

【 請求項2 】 前記ゲスト 計算機は、

専用のI / Oバスに接続され前記ゲスト 計算機においてのみアクセス可能な一群のゲスト 専用入出力装置と、

入出力命令で指定された入出力装置を識別する入出力装置識別情報を保持する入出力識別テーブルと、

を有し、

前記ゲスト プロセッサは、更に前記入出力識別テーブルを用いて入出力命令の対象となる入出力装置を判別することを特徴とする請求項1記載の複合計算機システム。

【 請求項3 】 前記エミュレーション手段は、前記ゲスト プロセッサが前記ホスト 計算機のいずれかの入出力装置に対する入出力命令を受けたときに実行するチャンネルプログラムに基づき前記入出力装置のアドレスと前記チャンネルプログラムのアドレスとを含む入出力要求情報を前記計算機間通信記憶領域に設定し、

前記ゲスト 計算機制御手段は、前記入出力要求情報に基 50

づき、指定された前記入出力装置に対する入出力命令を実行することを特徴とする請求項1 又は2 いずれかに記載の複合計算機システム。

【 請求項4 】 前記ゲスト プロセッサは、

作成された前記チャンネルプログラムを解釈する命令デコーダと、

機械語による実行指示内容に基づいて所定の処理を実行する実行ユニットと、

を有し、

前記実行ユニットは、前記命令デコーダの解釈に基づいた処理を行うことを特徴とする請求項1 又は2 いずれかに記載の複合計算機システム。

【 請求項5 】 前記ゲスト 制御手段は、前記ゲスト 計算機の状態を保持すると共に前記ゲスト 計算機から前記ホスト 計算機に対する割込みの主要因を保持するゲスト ステータスレジスタを有することを特徴とする請求項3 記載の複合計算機システム。

【 請求項6 】 前記ゲスト ステータスレジスタは、前記エミュレーション手段が前記ホスト 計算機に対して割込み処理の要求を行うことを示すビット情報を保持することを特徴とする請求項5 記載の複合計算機システム。

【 請求項7 】 前記ゲスト 制御手段は、前記ゲスト 計算機から前記ホスト 計算機に対して行う割込み処理要求を保持する割込み要求レジスタを有し、前記ホスト 制御手段は、前記割込み要求レジスタへの書き込みにより前記ゲスト 計算機からの割込み要求を認識することを特徴とする請求項3 記載の複合計算機システム。

【 請求項8 】 前記ゲスト 計算機制御手段は、前記ゲスト 計算機からの入出力要求を前記入出力装置のいずれかに対応付ける入出力マップテーブルを用いて、指定された前記入出力装置に対するマッピングの正当性を調べることを特徴とする請求項3 記載の複合計算機システム。

【 請求項9 】 前記入出力装置制御手段は、前記ゲスト 計算機制御手段の命令に従い、前記ゲスト 計算機が要求した入出力要求に基づく前記入出力装置と前記ゲスト 主記憶手段との間のデータ転送を、前記ホスト 計算機が提供するファイルシステムをそのまま利用して行うことを特徴とする請求項1 又は2 いずれかに記載の複合計算機システム。

【 請求項10 】 前記ホスト 計算機と前記ゲスト 計算機間の転送データの並び順を前記ホスト 計算機または前記ゲスト 計算機が取り扱うバイトデータ並びに変換するバイトオーダ変換機構を有することを特徴とする請求項1 又は2 いずれかに記載の複合計算機システム。

【 請求項11 】 前記ゲスト 制御手段は、前記ゲスト プロセッサに対する前記ホスト 計算機からの割込み要求を受け取る割込み受信レジスタを有することを特徴とする請求項1 又は2 いずれかに記載の複合計算機システム。

【 請求項12 】 前記割込み受信レジスタは、前記ホス

3

ト 計算機から入出力実行終了による割込み要求を保持するビット情報を保持し、
前記ゲストプロセッサは、そのビット情報を書込みにより前記入出力装置に対する入出力命令の実行の終了を認識することを特徴とする請求項1記載の複合計算機システム。

【請求項13】 前記入出力装置は、操作員が操作する端末を含み、
前記ゲスト計算機は、操作員インタフェース機能を実現する操作員エミュレーション手段を有し、前記端末から前記ゲスト計算機内のデータに対するアクセスを行わせることを特徴とする請求項1又は2いずれかに記載の複合計算機システム。

【請求項14】 前記端末に対応した前記入出力制御手段は、アクセスを行う前記ゲスト計算機内のデータのアドレス情報を指示内容として前記計算機間通信記憶領域に設定し、

前記操作員エミュレーション手段は、設定されたアドレス情報に従った処理を行うことを特徴とする請求項13記載の複合計算機システム。

【請求項15】 前記割込み受信レジスタは、前記ホスト計算機から操作員の指示による割込み要求を保持するビット情報を保持し、
前記ゲストプロセッサは、そのビット情報を書込むことにより前記ゲスト計算機に対する前記端末からのアクセスであることを認識することを特徴とする請求項11記載の複合計算機システム。

【請求項16】 第1の計算機と第2の計算機とから構成される複合計算機システムにおいて、

第1の計算機のオペレーティングシステムが第1の計算機の主記憶装置に格納され動作するステップと、

第1の計算機のオペレーティングシステムにおいて動作するインストーラが第1の計算機の主記憶装置に格納され動作するステップと、

前記インストーラにより第1の計算機に接続されたインストール媒体に格納されている第1の計算機で動作するソフトウェアと第2の計算機で動作するソフトウェアが第1の計算機と第2の計算機とで共用するディスク装置に格納されるステップと、

を含み、第2の計算機で動作するソフトウェアが第1の計算機のファイルとして認識される第1のヘッダ情報と第2の計算機のファイルとして認識される第2のヘッダ情報とを有することを特徴とするインストール方法。

【請求項17】 請求項1記載の複合計算機システムにおいて、

前記ホストプロセッサのオペレーティングシステムが動作するステップと、

前記ホストプロセッサのオペレーティングシステムにおいて動作するインストーラが動作するステップと、

前記インストーラによりホストプロセッサによりアクセ

4

スされるインストール媒体に格納されているホストプロセッサで動作するソフトウェアと、前記ゲストプロセッサで動作するソフトウェアと、が前記入出力装置に格納されるステップと、

を含み、前記ゲストプロセッサで動作するソフトウェアがホストプロセッサのファイルとして認識される第1のヘッダ情報とゲストプロセッサのファイルとして認識される第2のヘッダ情報とを有することを特徴とするインストール方法。

10 【請求項18】 前記入出力装置制御手段の動作を停止するステップと、

前記ホストプロセッサと独立したオペレーティングシステムの動作を停止するステップと、

前記ゲスト計算機制御手段の動作を停止するステップと、

インストールを実行するインストーラによる前記各ステップの停止処理の終了後にインストールを実行するステップと、

を含むことを特徴とする請求項17記載のインストール方法。

20

【請求項19】 請求項1記載の複合計算機システムにおいて、

前記入出力装置と前記入出力装置制御手段と前記ゲスト計算機制御手段に関する情報から構成されるシステム情報を参照するステップと、

前記入出力装置制御手段と前記ゲスト計算機制御手段に関する情報を検索し、対象ファイルを削除するステップと、

前記入出力装置に関する情報を検索し、前記ゲスト専用入出力装置に対応するファイルを削除するステップと、

前記システム情報を削除するステップと、

を含むことを特徴とするアンインストール方法。

【請求項20】 前記入出力装置制御手段の動作を停止するステップと、

前記ホストプロセッサと独立したオペレーティングシステムの動作を停止するステップと、

前記ゲスト計算機制御手段の動作を停止するステップと、

アンインストールを実行するアンインストールによる前記各ステップの停止処理の終了後にアンインストールを実行するステップと、

を含むことを特徴とする請求項19記載のアンインストール方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、クライアント／サーバシステムなど複数の計算機にある情報処理を分散して実行させる分散コンピュータシステムを構築する複合計算機システム、特に、ある計算機の処理動作の一部、入出力処理、システム操作員とのインタフェース処理等

50

を別の計算機を用いて実現するような柔軟性、高性能を追求するシステムの形態並びに複合計算機システムにおけるインストール方法及びアンインストール方法に関する。

【0002】

【従来の技術】従来のクライアント／サーバシステムは、サーバ機能を提供する計算機システムとクライアント機能を提供する計算機システムをネットワークで接続したシステムである。これは、例えばオーム社「ノベルのNet Ware 戦略」55ページの図3.2などに示されている。

【0003】一方、あるプロセッサの入出力処理を他方のプロセッサが代わって実行することで、ある計算機上で他機種のプロセッサ上で実行されるプログラムを実行可能にする計算機システムの従来例として、例えば特開平3-225551号公報がある。

【0004】図23は、特開平3-225551号公報において開示された入出力装置アクセス制御方式の構成例を示した図である。動作の詳細説明は特開平3-225551号公報の本文中に詳述されているので、こ

こでは省略する。

【0005】図23に示した従来の構成において、ゲストプロセッサ(図中のゲストCPU12を指す)において実行されるプログラムで入出力処理が発生すると、ホストプロセッサ(図中のホストCPU10を指す)に割込み要求を発生して、ホストプロセッサ上のエミュレータによりゲストプロセッサの入出力処理を実行する。こうすることで、計算機に他機種のプロセッサを接続して他機種プロセッサの計算機で実行するプログラムを実行可能にするシステムにおいて、他機種プログラムの実行環境の相違に容易に対応でき、プログラムの変更無く実行できるようになる効果が有る。

【0006】ところで、前述した計算機システムのように、一般にアーキテクチャの異なるクライアント計算機とサーバ計算機から構成されるシステムにおいては、ソフトウェアをそれぞれの計算機に接続されたディスク装置に格納するというインストール作業を行う場合、それぞれの計算機に用意されたインストーラがそれぞれのオペレーティングシステムおよびアプリケーションプログラムをインストールしていた。この従来のインストール方法について、以下に説明する。

【0007】図24は、従来のクライアント・サーバシステムのハードウェア構成図である。サーバ計算機は、プロセッサ31s、主記憶装置32s、バスのデータ転送制御及びプロセッサ31sに対する割込み制御等を行う制御装置33s、I/O装置34s及びI/O装置34sの一部である固定ディスク35sを有する。クライアント計算機も同様にプロセッサ31c、主記憶装置32c、制御装置33c、I/O装置34c及び固定ディスク35cを有する。

【0008】また、図25は、従来のクライアント・サーバシステムにインストールを行うために用いるインストール媒体を示した図である。インストール媒体としては、一般に図25に示したようにフロッピーディスクが用いられ、場合によってテープなども用いられる。インストール用のフロッピーディスクとして、サーバ計算機用、クライアント計算機用それぞれのオペレーティングシステム用インストール媒体(以下、OS用FD)51s、51cと、アプリケーションプログラム用インストール媒体(以下、アプリ用FD)52s、52cとが示されている。OS用FD51s、51cには、サーバ計算機用のオペレーティングシステム(以下、サーバOS)53s及びクライアント計算機用のオペレーティングシステム(以下、クライアントOS)53cと、サーバOS用のインストーラ54s及びクライアントOS用のインストーラ54cとがそれぞれ記憶されている。また、アプリ用FD52s、52cには、サーバ計算機用のアプリケーションプログラム(以下、サーバアプリケーション)55s及びクライアント計算機用のアプリケーションプログラム(以下、クライアントアプリケーション)55cと、サーバアプリケーション用のインストーラ56s及びクライアントアプリケーション用のインストーラ56cとがそれぞれ記憶されている。各OS53s、53c及び各アプリケーション55s、55cは、それぞれのインストーラ54s、54c及び56s、56cによってそれぞれの固定ディスク35s、35cにコピーされる。

【0009】インストール時には、まずそれぞれのOS用FD51に格納されているOSの一部あるいは全てをそれぞれの主記憶装置32に格納して動作させ、それぞれのOS上で動作するそれぞれのインストーラ54が、それぞれのOS53をOS用FD51からそれぞれの固定ディスク35にコピーする。また、それぞれの主記憶装置32に格納されて動作しているそれぞれのOS53は、それぞれの計算機のI/O装置34の構成情報を調べて、固定ディスク35のOSが管理する領域の一部に書き込む。次に、それぞれのOS53上で、それぞれのアプリケーション用インストーラ56が動作し、それぞれのアプリ用FD52から、それぞれのアプリケーションプログラム55をそれぞれの固定ディスク35にコピーする。また、アンインストーラ(図示せず)についても同様に、それぞれのアンインストーラがそれぞれのOSおよびアプリケーションプログラムをアンインストールしていた。

【0010】また、対象は異なるが、一つのインストーラで複数種類のOSをインストールできる従来例としては、特開平5-100828号公報に掲載されている技術があげられる。ここで開示されている技術は、あくまでも一台の計算機が対象であり、アーキテクチャが異なる複数のコンピュータを対象としているわけではない。

図26は上記特開平5 - 1 0 0 8 2 8 号公報に掲載されている技術を示す。図26において、コンピュータシステム41の主記憶装置(図示せず)に格納されたインストーラであるインストレーションプログラム43は、所定のフォーマットのインストレーション用媒体42に格納されたOSを、物理イメージで外部補助記憶装置44にコピーするものである。OSの種類が異なっているが、インストレーション用媒体42のフォーマットを全て同じにして、OSの種類だけインストレーション用媒体42を用意することで、1種類のインストレーションプログラム43で複数種類のOSをインストールできる効果がある。

【0011】

【発明が解決しようとする課題】しかしながら、従来のクライアント/サーバシステムでは、クライアントシステムとサーバシステムとは、共に独立した計算機システムであり、それぞれ別々のプロセッサ、メモリ、I/O装置を具備している。このようなクライアント・サーバシステムにおいては、最低でも2台の計算機システムが必要となり、高額の投資が必要になるという問題点を有している。

【0012】一方、特開平3 - 2 2 5 5 1 号公報に開示された入出力装置アクセス制御方式技術においては、計算機システムにおいて他機種のプロセッサをゲストプロセッサとして接続すると共に、制御部と、ホストプロセッサ上で動作するエミュレータとを付加することで、他機種のプロセッサ上で実行されるプログラムを何等変更することなく別の計算機システム上で実行することができる。しかし、従来においては、以下に示す複数の課題が存在する。

【0013】まず、ゲストプロセッサからの入出力処理要求をホストプロセッサに伝えると共にデータ転送及び動作制御を行う従来の制御部の構成では、ある時点でゲストプロセッサが実行可能な入出力処理は1つの入出力装置に対して1つのみである。この構成において、仮にゲストプロセッサが複数の入出力装置に同時並行して入出力処理を実行させようとする制御部のハードウェア量が多く必要になってしまう。

【0014】また、入出力処理を行うプログラムが動作するゲストプロセッサが複数の場合、技術的に対応可能かどうかが開示されていない。

【0015】また、ゲストプロセッサからの入出力処理要求に対する入出力データは、いったんゲストプロセッサを介して授受されることになるので、入出力チャネル機構のようなゲストプロセッサの命令実行と入出力処理とをゲストプロセッサ側において同時並行処理することができない。

【0016】更に、あるプロセッサが処理するデータのバイト単位での並び順等の変換を行う手段がないため、ゲストプロセッサとホストプロセッサが異機種である場

合、他機種のプロセッサの計算機で実行するプログラムを実行できないおそれが生じる。

【0017】また、アーキテクチャの異なるクライアント計算機とサーバ計算機から構成されるクライアント・サーバシステムへのインストールを行う方法であるが、前述したように、各計算機で動作するOSは、それぞれのアーキテクチャでのみ動作するが、それは、インストール時においても同じである。つまり、クライアントOSは、クライアント計算機の主記憶装置に格納されて動作し、サーバOSは、サーバ計算機の主記憶装置に格納されて動作する。そして、クライアント用のインストーラは、クライアントOSの上でのみ動作し、サーバ用のインストーラは、サーバOSの上でのみ動作する。また、インストール先となるI/O装置は、それぞれの計算機で異なるものであり、それらを制御するOSも上記のように異なるものである。すなわち、クライアント計算機用のI/O装置にソフトウェアをインストールするために用いるインストーラは、そのI/O装置を制御しているクライアントOS上で動作するプログラムである必要があり、同様にサーバ計算機用のインストーラは、サーバOS上で動作するプログラムである必要があった。

【0018】また、上記特開平5 - 1 0 0 8 2 8 号公報に掲載されているインストールに関する技術は、あくまでも単独の計算機システムに一つのOSをインストールするための技術である。また、論理フォーマットが異なる2つ以上のOSを一台の計算機にそれぞれ個別にインストールした場合のその計算機の動作については開示されていない。

【0019】すなわち、従来のインストール方法及びアンインストール方法では、一つのインストール媒体には一つのOSしか格納できないため、一度のインストール作業では一つのOSしかインストールできない。また、インストレーションプログラムは、論理フォーマットが異なるインストール媒体の内容をファイルとして認識できないため、物理イメージで外部補助記憶装置にコピーせざるを得ないという問題点があった。

【0020】また、第1の計算機と第2の計算機とから構成される複合計算機におけるインストール方法及びアンインストール方法についてはなんら開示されていない。

【0021】本発明は、上記課題を解決するためになされたものであり、その目的は、より高性能、低コストかつ柔軟なシステム構成が可能な複合計算機システムを提供することにある。

【0022】更に、本発明は、第1の計算機と第2の計算機とから構成される複合計算機において、一度の作業で複数のソフトウェアをインストール及びアンインストールでき、かつ第1の計算機からも第2の計算機からもファイルとして認識できるインストール方法及びアンインストール方法を提供することを目的とする。

【 0 0 2 3 】

【課題を解決するための手段】以上のような目的を達成するために、本発明は、ホスト 計算機と ゲスト 計算機とにより構成される複合計算機システムにおいて、前記ホスト 計算機は、プリ エンプティブ・ マルチタスク 処理機能を有するオペレーティングシステム及びアプリケーションプログラムが動作するホスト プロセッサと、システムバスに接続された一群の入出力装置と、前記入出力装置に対するアクセスを行うために前記入出力装置それぞれに対応して設けられ、前記ゲスト 計算機間のデータ転送を行う 入出力装置制御手段と、システムバス上のデータ転送制御及び前記ホスト プロセッサに対する割込み制御を行う ホスト 制御手段と、前記ゲスト 計算機を前記システムバスに接続された1 つの入出力装置とみなして前記ホスト 計算機における前記ゲスト 計算機の制御を一括して行う ゲスト 計算機制御手段と、前記ゲスト 計算機を接続するシステムバスと、を有し、前記ゲスト 計算機は、前記ホスト プロセッサと独立したオペレーティングシステム及びアプリケーションプログラムが動作するゲスト プロセッサと、前記ゲスト プロセッサによるエミュレーション命令を実行するエミュレーション手段と、前記システムバスを接続し、前記ホスト 計算機との間のデータ転送制御及び前記ゲスト プロセッサに対する割込み制御を行う ゲスト 制御手段と、前記ホスト 計算機からアクセス可能であり 各種情報を相互に交換する 計算機間通信記憶領域を含むゲスト 主記憶手段とを有し、前記ホスト 計算機の入出力装置に対するアクセスを前記ゲスト プロセッサを用いることなく行うことを特徴とする。

【 0 0 2 4 】また、前記ゲスト 計算機は、専用のI / Oバスに接続され前記ゲスト 計算機においてのみアクセス可能な一群のゲスト 専用入出力装置と、入出力命令で指定された入出力装置を識別する入出力装置識別情報を保持する入出力識別テーブルと、を有し、前記ゲスト プロセッサは、更に前記入出力識別テーブルを用いて入出力命令の対象となる入出力装置を判別することを特徴とする。

【 0 0 2 5 】また、前記エミュレーション手段は、前記ゲスト プロセッサが前記ホスト 計算機のいずれかの入出力装置に対する入出力命令を受けたときに実行するチャネルプログラムに基づき前記入出力装置のアドレスと前記チャネルプログラムのアドレスとを含む入出力要求情報を前記計算機間通信記憶領域に設定し、前記ゲスト 計算機制御手段は、前記入出力要求情報に基づき、指定された前記入出力装置に対する入出力命令を実行することを特徴とする。

【 0 0 2 6 】また、前記ゲスト プロセッサは、作成された前記チャネルプログラムを解釈する命令デコーダと、機械語による実行指示内容に基づいて所定の処理を実行する実行ユニットとを有し、前記実行ユニットは、前記命令デコーダの解釈に基づいた処理を行うことを特徴と

する。

【 0 0 2 7 】また、前記ゲスト 制御手段は、前記ゲスト 計算機の状態を保持すると共に前記ゲスト 計算機から前記ホスト 計算機に対する割込みの主要因を保持するゲスト ステータスレジスタを有することを特徴とする。

【 0 0 2 8 】また、前記ゲスト ステータスレジスタは、前記エミュレーション手段が前記ホスト 計算機に対して割込み処理の要求を行うことを示すビット 情報を保持することを特徴とする。

【 0 0 2 9 】また、前記ゲスト 制御手段は、前記ゲスト 計算機から前記ホスト 計算機に対して行う 割込み処理要求を保持する割込み要求レジスタを有し、前記ホスト 制御手段は、前記割込み要求レジスタへの書込みにより前記ゲスト 計算機からの割込み要求を認識することを特徴とする。

【 0 0 3 0 】また、前記ゲスト 計算機制御手段は、前記ゲスト 計算機からの入出力要求を前記入出力装置のいずれかに対応付ける入出力マップテーブルを用いて、指定された前記入出力装置に対するマッピングの正当性を調べることを特徴とする。

【 0 0 3 1 】また、前記入出力装置制御手段は、前記ゲスト 計算機制御手段の命令に従い、前記ゲスト 計算機が要求した入出力要求に基づく 前記入出力装置と前記ゲスト 主記憶手段との間のデータ転送を、前記ホスト 計算機が提供するファイルシステムをそのまま利用して行うことを特徴とする。

【 0 0 3 2 】また、前記ホスト 計算機と前記ゲスト 計算機間の転送データの並び順を前記ホスト 計算機または前記ゲスト 計算機が取り扱う バイト データ並びに変換する バイト オーダ変換機構を有することを特徴とする。

【 0 0 3 3 】また、前記ゲスト 制御手段は、前記ゲスト プロセッサに対する前記ホスト 計算機からの割込み要求を受け取る割込み受信レジスタを有することを特徴とする。

【 0 0 3 4 】また、前記割込み受信レジスタは、前記ホスト 計算機から入出力実行終了による割込み要求を保持するビット 情報を保持し、前記ゲスト プロセッサは、そのビット 情報を書込みにより前記入出力装置に対する入出力命令の実行の終了を認識することを特徴とする。

【 0 0 3 5 】また、前記入出力装置は、操作員が操作する端末を含み、前記ゲスト 計算機は、操作員インタフェース機能を実現する操作員エミュレーション手段を有し、前記端末から前記ゲスト 計算機内のデータに対するアクセスを行わせることを特徴とする。

【 0 0 3 6 】また、前記端末に対応した前記入出力制御手段は、アクセスを行う 前記ゲスト 計算機内のデータのアドレス情報を指示内容として前記計算機間通信記憶領域に設定し、前記操作員エミュレーション手段は、設定されたアドレス情報に従った処理を行うことを特徴とする。

【 0 0 3 7 】更に、前記割込み受信レジスタは、前記ホスト 計算機から操作員の指示による割込み要求を保持するビット 情報を保持し、前記ゲスト プロセッサは、そのビット 情報を書込みにより 前記ゲスト 計算機に対する前記端末からのアクセスであることを認識することを特徴とする。

【 0 0 3 8 】また、本発明に係るインストール方法は、第1 の 計算機と第2 の 計算機とから構成される複合計算機システムにおいて、第1 の 計算機のオペレーティングシステムが第1 の 計算機の主記憶装置に格納され動作するステップと、第1 の 計算機のオペレーティングシステムにおいて動作するインストーラが第1 の 計算機の主記憶装置に格納され動作するステップと、前記インストーラにより 第1 の 計算機に接続されたインストール媒体に格納されている第1 の 計算機で動作するソフトウェアと第2 の 計算機で動作するソフトウェアが第1 の 計算機と第2 の 計算機とで共用するディスク装置に格納されるステップとを含み、第2 の 計算機で動作するソフトウェアが第1 の 計算機のファイルとして認識される第1 のヘッダ情報と第2 の 計算機のファイルとして認識される第2 のヘッダ情報とを有することを特徴とする。

【 0 0 3 9 】また、第1 の発明に係る複合計算機システムにおいて、前記ホスト プロセッサのオペレーティングシステムが動作するステップと、前記ホスト プロセッサのオペレーティングシステムにおいて動作するインストーラが動作するステップと、前記インストーラによりホスト プロセッサによりアクセスされるインストール媒体に格納されているホスト プロセッサで動作するソフトウェアと、前記ゲスト プロセッサで動作するソフトウェアと、が前記入出力装置に格納されるステップとを含み、前記ゲスト プロセッサで動作するソフトウェアがホスト プロセッサのファイルとして認識される第1 のヘッダ情報とゲスト プロセッサのファイルとして認識される第2 のヘッダ情報とを有することを特徴とする。

【 0 0 4 0 】また、上記インストール方法において、前記入出力装置制御手段の動作を停止するステップと、前記ホスト プロセッサと独立したオペレーティングシステムの動作を停止するステップと、前記ゲスト 計算機制御手段の動作を停止するステップと、インストールを実行するインストーラによる前記各ステップの停止処理の終了後にインストールを実行するステップとを含むことを特徴とする。

【 0 0 4 1 】また、本発明に係るアンインストール方法は、第1 の発明に係る複合計算機システムにおいて、前記入出力装置と前記入出力装置制御手段と前記ゲスト 計算機制御手段に関する情報から構成されるシステム情報を参照するステップと、前記入出力装置制御手段と前記ゲスト 計算機制御手段に関する情報を検索し、対象ファイルを削除するステップと、前記入出力装置に関する情報を検索し、前記ゲスト 専用入出力装置に対応するファ

イルを削除するステップと、前記システム情報を削除するステップとを含むことを特徴とする。

【 0 0 4 2 】また、上記アンインストール方法において、前記入出力装置制御手段の動作を停止するステップと、前記ホスト プロセッサと独立したオペレーティングシステムの動作を停止するステップと、前記ゲスト 計算機制御手段の動作を停止するステップと、アンインストールを実行するアンインストーラによる前記各ステップの停止処理の終了後にアンインストールを実行するステップとを含むことを特徴とする。

【 0 0 4 3 】
【 発明の実施の形態】以下、図面に基づいて、本発明の好適な実施の形態について説明する。

【 0 0 4 4 】本実施の形態におけるシステム構成
図1 は、本発明に係る複合計算機システムの一実施例を示す概略的なブロック構成図である。図1 に示したように本システムは、ホスト 計算機1 0 0 とゲスト 計算機3 0 0 とをホスト 計算機1 0 0 におけるシステムバス1 3 0 で接続した構成となっている。なお、本実施の形態において、ホスト 計算機1 0 0 側の構成要素は1 0 0 番台の符号を、ゲスト 計算機3 0 0 側の構成要素は3 0 0 番台の符号を、それぞれ付ける。

【 0 0 4 5 】ホスト 計算機1 0 0 は、各種プログラムやデータを記憶するホスト 主記憶装置1 1 0 と、ホスト 計算機1 0 0 における中央処理装置であるホスト プロセッサ1 2 0 と、システムバス1 3 0 上のデータ転送制御及びホスト プロセッサ1 2 0 に対する割込み制御を行うホスト 制御手段としてのホスト 制御装置1 4 0 と、システムバス1 3 0 に接続されホスト 計算機1 0 0 における一群の入出力装置であるホスト I / O 装置1 5 0 とを有する。

【 0 0 4 6 】ゲスト 計算機3 0 0 は、プログラムやデータを記憶するゲスト 主記憶装置3 1 0 と、ゲスト 計算機3 0 0 における中央処理装置でありホスト プロセッサ1 2 0 と独立したオペレーティングシステム及びアプリケーションプログラムが動作するゲスト プロセッサ3 2 0 と、システムバス1 3 0 を接続し、ホスト 計算機1 0 0 との間のデータ転送制御及びゲスト プロセッサ3 2 0 に対する割込み制御を行うゲスト 制御手段としてのゲスト 制御装置3 3 0 と、各装置を接続するゲスト 主記憶バス3 4 0 と、ゲスト プロセッサ3 2 0 に接続されたゲスト 計算機3 0 0 独自の専用のI / O バス3 5 0 と、I / O バス3 5 0 に接続されゲスト 計算機3 0 0 のみがアクセス可能な一群のゲスト 専用入出力(I / O) 装置3 6 0 とを有する。ゲスト 専用入出力(I / O) 装置3 6 0 は、少なくとも1 台の入出力装置によって構成される。

【 0 0 4 7 】図2 は、本実施の形態における各計算機で実行されるソフトウェア構成の一実施例を示した図である。

【 0 0 4 8 】ホスト 計算機1 0 0 には、ホスト 計算機1

00においてシステムリソースを管理、制御すると共に
 プリエンティブ・マルチタスク処理によりアプリケーションプログラムの強制切換え機能を有するホストオペレーティングシステム(以下ホストOSと称す)111及び次の各種アプリケーションプログラムが用意されている。ゲスト計算機ドライバ112は、システムバス130に接続されたゲスト計算機300をホスト計算機100から見て1つの入出力装置として扱い、ホスト計算機100におけるゲスト計算機300のハードウェア動作を一括して制御する。I/O装置制御プログラム113は、接続された入出力装置毎に用意されており、ゲスト計算機ドライバ112により起動されて各入出力装置に対する入出力動作を制御する。詳細は後述するが、具体的には、ゲスト計算機300から発行された入出力(I/O)命令実行要求により特定された個別の入出力装置とゲスト主記憶装置との間のデータ転送等を制御する。操作員インタフェース制御プログラム114は、ホストOS111を経由して操作員が操作するホストI/O装置150の入出力装置とゲスト計算機300との間で通信を行うことでゲスト計算機300において必要とされる操作員機能を実現する。アプリケーションプログラム115は、ホスト計算機100のホストOS111上で動作する各種のプログラムである。以上の説明から理解できるように、本実施の形態における入出力装置制御手段はI/O装置制御プログラム113及び操作員インタフェース制御プログラム114を、ゲスト計算機制御手段はゲスト計算機ドライバ112を、それぞれホストプロセッサ120で実行させることにより実現される。

【0049】一方、ゲスト計算機300には、ゲスト計算機300においてシステムリソースを管理制御しアプリケーションプログラムの実行を可能にするゲストオペレーティングシステム(以下ゲストOSと称す)311と、ゲスト主記憶装置310内の特定領域に配置されて、ゲストプロセッサ320の機械語命令セットを用いたプログラムによりエミュレーション命令を実行することでゲスト計算機300の入出力命令実行機能や操作員インタフェース機能を実現するエミュレーションプログラム312と、ゲストOS311上で動作する各種のアプリケーションプログラム313である。以上の説明から理解できるように、本実施の形態におけるエミュレーション手段は、ゲスト計算機300の入出力命令実行機能を実現するエミュレーションプログラム312を、また、操作員エミュレーション手段は、操作員インタフェース機能を実現するエミュレーションプログラム312を、それぞれゲストプロセッサ320で実行させることにより実現される。このように、エミュレーションプログラム312は、実行する機能により異なる機能のプログラムが選択され実行されることになる。

【0050】図3は、図1及び図2で示された本実施の

形態における複合計算機システムのハードウェア、ソフトウェア構成例の動作を説明するシステム構成図である。なお、図1及び図2と同じ要素には同じ符号を付ける。

【0051】ゲスト計算機300におけるゲスト主記憶手段としてのゲスト主記憶装置310は、図2に示したソフトウェアが記憶される以外に、ゲスト計算機300においてデータ転送を実行するときに使用され実際の入出力動作を規定するチャンネルプログラム314を記憶する。また、特定アドレス領域に設けられてゲストプロセッサ320及びホストプロセッサ120から共有され相互にアクセス可能な計算機間通信記憶領域315が設けられ、ここには、ゲスト計算機300からホスト計算機100に対する割込み要求に付随する詳細情報、ホスト計算機100からゲスト計算機300への割込み要求時の割込み要因などの情報、I/O装置制御プログラム113とエミュレーションプログラム312との間の同期フラグ等の各種情報が記憶され、相互に交換される。図4にこの内部構成例を示すが、この詳細は動作例において適宜説明する。

【0052】ゲストプロセッサ320は、命令デコーダ321、実行ユニット322及び入出力識別テーブル323を搭載する。命令デコーダ321は、チャンネルプログラム等の機械語命令を解釈し実行の準備を行う。実行ユニット322は、命令デコーダ321から供給される実行指示内容に基づいて機械語命令の実行を行う。また、機械語命令の実行において、エミュレーションを行わせるためのエミュレーション命令かどうかを判断し、エミュレーション命令が検出された場合にはエミュレーションを実行し、ゲスト専用I/O装置360への入出力命令の場合は、I/Oバス350経由で入出力を行い、あるいは割込み発生時には割込み処理を実行する。入出力識別テーブル323は、入出力命令で指定されたI/O装置がホストI/O装置150かゲスト専用I/O装置360かを識別するための識別情報を保持しており、ゲストプロセッサ320によって用いられる。この入出力識別テーブル323の例を図5に示す。

【0053】ゲスト制御装置330は、システムバスインタフェース制御部(以下、「システムバスI/F制御部」と称す)331、バイトオーダ変換機構332、割込み要求レジスタ333、ゲストステータスレジスタ334及び割込み受信レジスタ335を有する。システムバスI/F制御部331は、ゲスト計算機300からシステムバス130へのインタフェース制御やデータ転送動作を制御する。バイトオーダ変換機構332は、ホスト主記憶装置110上で処理単位内のバイトデータの並びと、ゲスト主記憶装置310上で処理単位内のバイトデータの並び、すなわちバイトオーダが同じ場合にはホスト計算機100からゲスト主記憶装置310への全てのアクセス時に何の変換も行わず、ホスト計算機100

15

0 とゲスト 計算機3 0 0 のバイトオーダが異なる場合には、ホスト 計算機1 0 0 からゲスト 主記憶装置3 1 0 への全てのアクセス時にゲスト 計算機3 0 0 におけるバイトデータ並びに合わせた変換を行う。バイトオーダ変換機構3 3 2 の構成及び動作についての詳細は後述する。割り込み要求レジスタ3 3 3 は、ゲスト 計算機3 0 0 からホスト 計算機1 0 0 に対して行う割り込み処理要求を保持し、ホスト 計算機1 0 0 が備える割り込み中の1 割り込み要求レベルとしてホスト 制御装置1 4 0 経由でホスト プロセッサ1 2 0 に通知する。この割り込み要求レジスタ3 3 3 は、ゲストステータスレジスタ3 3 4 に含まれるいずれかのビット情報がセットされると、セットされるので、ホスト プロセッサ1 2 0 は、ホスト 計算機1 0 0 で割り込みが発生したことを即座に知ることができる。ゲストステータスレジスタ3 3 4 は、ゲスト 計算機3 0 0 の状態を保持すると共にゲスト 計算機3 0 0 からホスト 計算機1 0 0 に対する割り込みの主要因を保持する。従って、ゲストステータスレジスタ3 3 4 の内容を参照すれば、割り込みの発生要因を知ることができる。図6 にこの内容を示すが、詳細は後述する。割り込み受信レジスタ3 3 5 は、ゲストプロセッサ3 2 0 に対するホスト 計算機1 0 0 からの割り込み要求を受け取る。図7 にこの内容を示すが、詳細は後述する。

【0054】また、ホスト 計算機1 0 0 に接続された入出力装置として、本実施の形態においては、固定ディスク装置(以下「HDD」と称す)1 5 1、磁気テープ装置(以下「MT」と称す)1 5 2、ローカルエリアネットワーク制御装置(以下「LAN」と称す)1 5 3、ホスト 計算機1 0 0 及びゲスト 計算機3 0 0 両方に対して操作員が使用する入出力装置(以下「端末」と称す)1 5 4 及び上記以外の例えばフロッピーディスク装置やCD-ROM装置等他のI/O装置1 5 5 を搭載する。他のI/O装置1 5 5 は、必ずしもゲスト 計算機3 0 0 をサポートする必要はないものとする。ホスト 主記憶装置1 1 0 に記憶されるI/O装置制御プログラム1 1 3 は、上記各入力装置に対応させて、I/O装置制御プログラム(HDD)1 1 3 a、I/O装置制御プログラム(MT)1 1 3 b、I/O装置制御プログラム(LAN)1 1 3 c 及び操作員インタフェース制御プログラム1 1 4 等をそれぞれ記憶することになる。

【0055】本実施の形態におけるホスト 計算機1 0 0 は、主記憶装置、プロセッサ、入出力装置を全て具備して完全なコンピュータシステムを構成すると共に、ホスト I/O装置の一部あるいは全てをゲスト 計算機3 0 0 の入出力装置と共用できるような構成を有している。

【0056】本実施の形態におけるゲスト 計算機3 0 0 は、主記憶装置、プロセッサを具備するが、入出力装置及び操作員インタフェース装置をホスト 計算機1 0 0 の入出力装置と共用することで、本来のゲスト 計算機3 0 0 と同じ完全なコンピュータシステムを構成する。

16

【0057】また、本実施の形態におけるホスト 計算機1 0 0 のシステムバス1 3 0 は、ホスト 主記憶装置1 1 0 とホスト I/O装置1 5 0 の間でのデータ転送経路として用いられると共に、ゲスト 計算機3 0 0 のゲスト 主記憶装置3 1 0 とホスト I/O装置1 5 0、更にホスト プロセッサ1 2 0 とゲスト プロセッサ3 2 0 の間でのデータ転送経路としても用いられる。

【0058】図6 は、前述したゲストステータスレジスタ3 3 4 の構成例であり、以下に示すビット情報を保持する。図6 において、HWI ビットは、ゲスト 計算機3 0 0 のハードウェアがホスト 計算機1 0 0 に対して割り込み処理の要求を行うことを意味しており、例えばゲスト 計算機3 0 0 中のハードウェアエラーが発生した場合などにハードウェアによりセットされる。EMI ビットは、エミュレーションプログラム3 1 2 がホスト 計算機1 0 0 に対して割り込み処理の要求を行うことを意味しており、例えば入出力命令の実行時にホスト 計算機1 0 0 に入出力処理の起動要求を行う場合などにセットされる。STI 1 ビットは、ゲスト 計算機3 0 0 においてゲスト プロセッサ3 2 0 の状態変化が発生したことを意味しており、特にゲスト プロセッサ3 2 0 が停止(stop)状態に入ったことを示している。STI 2 ビットは、ゲスト 計算機3 0 0 においてゲスト プロセッサ3 2 0 の状態変化が発生したことを意味しており、特にゲスト プロセッサ3 2 0 が命令実行(operating)状態に入ったことを示している。STI 1 ビット及びSTI 2 ビットは、共にゲスト 計算機3 0 0 のハードウェアによりセットされる。GPROC Status は、ゲスト プロセッサ3 2 0 の現在状態を表示するビット群であり、ゲスト プロセッサ3 2 0 が直接セットする。HWI、EMI、STI 1、STI 2 のうちのいずれか少なくとも1 ビットがセットされると、割り込み要求レジスタ3 3 3 がセットされて、システムバス1 3 0 経由でホスト 制御装置1 4 0 に対してゲスト 計算機3 0 0 からの割り込み要求が伝えられたことになる。

【0059】図7 は、ゲスト 制御装置3 3 0 内にある割り込み受信レジスタ1 6 5 の内部構成を示した図である。割り込み受信レジスタ1 6 5 は、ホスト 計算機1 0 0 からの割り込み要求を検出するためのレジスタであり、IOI ビットとOPI ビットとのビット情報を保持する。IOI ビットは、入出力終了割り込み要求の場合にセットされる。OPI ビットは、操作員からの指示によるゲスト 計算機3 0 0 に対する操作員インタフェース処理割り込み要求の場合にセットされる。

【0060】図8 は、ホスト 計算機1 0 0 においてゲスト 計算機3 0 0 から発行された入出力要求をホスト I/O装置1 5 0 の1 つに対応付けるI/Oマップ機構が用いる入出力マップテーブル1 1 6 の構成例を示した図である。これは、ホスト 主記憶装置1 1 0 に記憶される。

【0061】図9 は、前述したバイトオーダ変換機構3

3 2 の構成例を示した図である。バイト オーダ変換機構 3 3 2 は、システムバス 1 3 0 上の 2 つのバイト 位置データ入力信号から選択信号の値に従ってどちらか 1 つをゲスト 主記憶バス 3 4 0 上に出力信号として選択するマルチプレクサ 3 3 6 と、ゲスト 主記憶バス 3 4 0 上の 2 つのバイト 位置データ入力信号から選択信号の値に従ってどちらか 1 つをシステムバス 1 3 0 上に出力信号として選択するマルチプレクサ 3 3 7 と、ゲスト 主記憶バスに対して出力制御信号が出力可能を示す値の時には入力信号をゲスト 主記憶バス上に出力し、ゲスト 主記憶バスに対して出力制御信号が出力可能を示す値の時にはゲスト 主記憶バス上に出力しないように動作するバッファ 3 3 8 と、システムバス 1 3 0 に対して出力制御信号が出力可能を示す値の時には入力信号をシステムバス 1 3 0 上に出力し、システムバス 1 3 0 に対して出力制御信号が出力不可能を示す値の時にはシステムバス 1 3 0 上に出力しないように動作するバッファ 3 3 9 とを有する。

【 0 0 6 2 】 図 1 0 は、ゲスト 計算機 3 0 0 において実行される入出力命令において指定された HDD 1 5 1 に対する入出力動作をホスト 計算機 1 0 0 上のファイルにマッピングする時の動作を示す図である。

【 0 0 6 3 】 本実施の形態における動作の説明

本実施の形態において特徴的なことは、ホスト I / O 装置 1 5 0 をゲスト 計算機 3 0 0 からも共用できるようにしたことであり、かつゲスト 計算機 3 0 0 が行うホスト I / O 装置 1 5 0 に対する入出力処理をゲスト 計算機 3 0 0 のゲスト プロセッサ 3 2 0 を用いることなく行うことができるようにしたことである。この本実施の形態における特徴的な動作を、(1) ゲスト 計算機 3 0 0 の入出力命令実行動作、(2) ゲスト 計算機 3 0 0 の入出力命令で指定されたデータ転送動作、(3) ゲスト 計算機 3 0 0 の入出力終了割込み処理動作、(4) ゲスト 計算機 3 0 0 の専用 I / O 装置 3 6 0 への入出力命令実行動作、(5) 操作員とゲスト 計算機 3 0 0 とのコミュニケーション動作及び(6) インストール/アンインストール方法、に分けて説明する。

【 0 0 6 4 】 (1) ゲスト 計算機 3 0 0 の入出力命令実行動作

図 1 1 及び図 1 2 は、本実施の形態においてゲスト 計算機 3 0 0 中の入出力命令を前述した複合計算機システムにおいて実行する時の内部処理フローを示した図であり、図 1 2 は図 1 1 の続きの図である。

【 0 0 6 5 】 まず、最初にゲスト 計算機 3 0 0 における入出力命令の実行について、図 3 、図 1 1 及び図 1 2 に基づいて説明する。

【 0 0 6 6 】 ゲスト 計算機 3 0 0 のアプリケーションプログラム 3 1 3 において、ファイルのアクセス (例えば read や write) が実行されると、アプリケーションプログラム 3 1 3 はゲスト OS 3 1 1 に対してシステムコールを行う。システムコールで呼び出されたゲス

ト OS 3 1 1 は、アプリケーションプログラム 3 1 3 から指定されたファイルを、ゲスト OS 3 1 1 が管理するファイルシステムを通してファイルが格納される固定ディスク装置上の物理位置に変換する。この固定ディスク装置は、ホスト I / O 装置 1 5 0 との間でデータ転送を行うために用いられるゲスト 計算機 3 0 0 上の仮想的な装置である。

【 0 0 6 7 】 次に、ゲスト OS 3 1 1 は、ハードウェアに入出力動作を行わせるために、I / O 装置制御プログラム 1 1 3 が解釈及び実行できる形式であるチャンネルプログラム 3 1 4 をゲスト 主記憶装置 3 1 0 上に準備する。チャンネルプログラム 3 1 4 は、例えば図 1 0 に示したように、入出力装置からゲスト 主記憶装置 3 1 0 に対するデータの読み込み (read) やゲスト 主記憶装置 3 1 0 から入出力装置に対してデータの書き出し (write) といった動作の種類を指定するコマンドと、コマンドで指定される動作で使用するシリンダ番号やトラック番号といったより詳細な入出力装置に対する情報とから構成されるチャンネルコマンドワードを複数組み合わせで構成する。

【 0 0 6 8 】 チャンネルプログラム 3 1 4 の作成が完了すると、ゲスト OS 3 1 1 は、ファイルアクセスの対象となるいずれかの入出力装置を指定して入出力動作を開始させるゲスト 入出力命令を実行する。

【 0 0 6 9 】 ゲスト 入出力命令の機械語形式は、例えば図 1 0 に示したように入出力命令を特定する命令コードと、I / O 装置アドレス情報を含む語とから構成される。I / O 装置アドレス情報というのは、入出力動作の対象つまり転送相手となる I / O 装置を特定するための情報である。本実施の形態においては、ホスト I / O 装置 1 5 0 及びゲスト 専用 I / O 装置 3 6 0 が I / O 装置として設けられているので、いずれかに属する入出力装置そのもの又は入出力装置上のファイルを特定するための情報が設定されている。ホスト I / O 装置 1 5 0 に関する I / O 装置アドレス情報の内容は、ホスト 計算機 1 0 0 とともに予め知らされており、ホスト 計算機 1 0 0 においては入出力マップテーブル 1 1 6 に登録されている。

【 0 0 7 0 】 ゲスト プロセッサ 3 2 0 中の命令デコーダ 3 2 1 において解釈された入出力命令は、実行ユニット 3 2 2 に送られて入出力命令の実行が開始される。実行ユニット 3 2 2 は、入出力命令に含まれる I / O 装置アドレス情報と入出力識別テーブル 3 2 3 の識別情報に基づいて、命令デコーダ 3 2 1 から受け取った入出力命令をエミュレーションプログラム 3 1 2 で実行するエミュレーション命令であると判断すると、実行状態をエミュレーションプログラム実行状態に切り換えた後、エミュレーションプログラム 3 1 2 を呼び出すという所定の処理を行う。なお、入出力命令をゲスト 専用 I / O 装置 3 6 0 に対する命令と判断した場合の処理の詳細は、後述

する。

【 0 0 7 1 】 呼び出されたエミュレーションプログラム 3 1 2 は、入出力命令の実行処理を開始する。まず、エミュレーションプログラム 3 1 2 は、入出力命令による入出力処理の開始をホスト 計算機 1 0 0 に伝える準備として、計算機間通信記憶領域 3 1 5 にある G F A C T O R に入出力命令の実行要求及び入出力処理の動作の対象となるホスト I / O 装置 1 5 0 の I / O 装置アドレスを、G D A T A にゲスト 主記憶装置 3 1 0 でチャンネルプログラム 3 1 4 が格納されている先頭アドレス値を、そ
10 それぞれ入出力要求情報としてセットする。

【 0 0 7 2 】 続いてエミュレーションプログラム 3 1 2 は、ホスト 計算機 1 0 0 に対して入出力命令処理要求を伝えるために、図 6 に示したゲスト ステータスレジスタ 3 3 4 に含まれるホスト 計算機 1 0 0 に対する割り込み処理要求を意味する E M I ビットをセットする。E M I ビットをセットした後、エミュレーションプログラム 3 1 2 は、ホスト 計算機 1 0 0 による入出力命令実行が完了して計算機間通信記憶領域 3 1 5 の中にある G T E R M ビットがホスト 計算機 1 0 0 によりセットされるまでポー
20 ーリングしながら待ちつづける。

【 0 0 7 3 】 このように、ゲスト 計算機 3 0 0 にエミュレーション手段を設けたので、ゲスト プロセッサ 3 2 0 上で実行される入出力命令を実際に実行することができ、ゲスト プロセッサ 3 2 0 の機械語命令セット及びハードウェアの変更が不要となる。

【 0 0 7 4 】 次に、ゲスト ステータスレジスタ 3 3 4 の E M I ビットがセットされると、割り込み要求レジスタ 3 3 3 がセットされて、システムバス 1 3 0 を介してホスト 計算機 1 0 0 のホスト 制御装置 1 4 0 にゲスト 計算機 3 0 0 からの割り込み要求が伝えられる。ホスト 制御装置 1 4 0 は、ゲスト 計算機 3 0 0 からの割り込み要求を認識すると、ホスト プロセッサ 1 2 0 に対して割り込み要求を発生する。ホスト プロセッサ 1 2 0 は、割り込み受け可能状態になると、ゲスト 計算機 3 0 0 からの割り込み要求を受け付けて、割り込み処理に入る。

【 0 0 7 5 】 ゲスト 計算機 3 0 0 からの割り込み要求を受け付けたホスト プロセッサ 1 2 0 は、ゲスト 計算機 3 0 0 からの割り込み要求であることを認識すると、ゲスト 計算機ドライバ 1 1 2 を起動して割り込み処理を開始する。
40 ゲスト 計算機ドライバ 1 1 2 は、ゲスト 計算機 3 0 0 から要求された割り込みの要因を知るために、最初にゲスト ステータスレジスタ 3 3 4 を読み出す。この時にゲスト ステータスレジスタ 3 3 4 の割り込み要求を発生させるビット 群つまり H W I 、 E M I 、 S T I 1 、 S T I 2 ビット及び割り込み要求レジスタ 3 3 3 はリセットされる。ゲスト 計算機ドライバ 1 1 2 は、読み取ったゲスト ステータスレジスタ 3 3 4 の中を調べて E M I ビットがセットされていることからエミュレーションプログラム 3 1 2 による割り込み要求であることを認識する。

【 0 0 7 6 】 続いてゲスト 計算機ドライバ 1 1 2 は、エミュレーションプログラム 3 1 2 による割り込み要求の詳細要因を調べるために、計算機間通信記憶領域 3 1 5 中の G F A C T O R を読みだす。G F A C T O R には上述の通りエミュレーションプログラム 3 1 2 によりセットされた入出力命令の実行要求と入出力処理の動作の対象となる入出力装置の I / O 装置アドレスが格納されていることから、ゲスト 計算機ドライバ 1 1 2 は、入出力命令実行と指定されたホスト I / O 装置 1 5 0 に対する入出力処理動作の起動を行う必要があることを認識する。更に、ゲスト 計算機ドライバ 1 1 2 は、計算機間通信記憶領域 3 1 0 中の G D A T A を読みだし、ゲスト 主記憶装置 3 1 0 中でチャンネルプログラム 3 1 4 が格納されている先頭アドレス情報を得る。

【 0 0 7 7 】 次に、ゲスト 計算機ドライバ 1 1 2 は、図 8 に示された I / O マップ機構のマップテーブル 1 1 6 を用いて、I / O 装置アドレスに対応した転送相手にマッピングできるかなどの正当性をチェックする。図 8 に示した「ゲスト 計算機での I / O 装置アドレス」というのは、ゲスト O S 3 1 1 によって実行された入出力命令に含まれている I / O 装置アドレスであり、転送相手となるのは、ホスト I / O 装置 1 5 0 に含まれるいずれかの入出力装置そのもの又は入出力装置に含まれるファイルである。

【 0 0 7 8 】 図 1 0 に示したゲスト 入出力命令に含まれる I / O 装置アドレスが、図 8 に示したように例えば 1 6 進数表記で x ' 1 0 で指定された I / O 装置アドレス I O A であつたとすると、マップテーブル 1 1 6 を用いることで $I O A = x ' 1 0$ に対応するホスト 計算機 1 0 0 の H D D 1 5 1 中のファイル名 D : ¥ G I O ¥ H D D 0 1 を得ることができる。この時点で、ゲスト 計算機ドライバ 1 1 2 は、ゲスト 計算機 3 0 0 からの入出力要求における転送相手を特定することができる。なお、入出力マップテーブル 1 1 6 を用いることで、I / O 装置アドレス I O A というようなアドレス値を用いなくても単なる識別子を用いてもよいことがわかる。

【 0 0 7 9 】 もし、ゲスト O S 3 1 1 が指定した I / O 装置アドレスが、ホスト 計算機 1 0 0 において提供されないなどの未定義あるいは不正装置アドレスであつた場合や、指定されたホスト I / O 装置 1 5 0 が既に別の入出力命令により動作中である場合など H D D 1 5 1 における入出力動作が実行不可能であると判断した場合には、ゲスト 計算機ドライバ 1 1 2 は処理結果を G T E R M にセットして割り込み処理を終了させる。

【 0 0 8 0 】 一方、ゲスト 計算機ドライバ 1 1 2 において G F A C T O R の内容から入出力命令が指定する I / O 装置アドレスに対応するホスト 計算機 1 0 0 のホスト I / O 装置 1 5 0 (現在説明している例では H D D 1 5 1 上のファイル D : ¥ G I O ¥ H D D 0 1) による入出力動作が実行可能であると判断すると、ゲスト 計算機ド
50

ライバ1 1 2 は、入出力命令が指定するI / O 装置アドレス即ちHDD1 5 1 に対応するI / O 装置制御プログラム(HDD) 1 1 3 a に対して入出力動作の開始を指示すると共に、計算機間通信記憶領域3 1 5 にセットされたGFACTOR、GDATA 情報、及び入出力マップテーブル1 1 6 に基づくファイル名情報を引き渡す。I / O 装置制御プログラム(HDD) 1 1 3 a は、GDATA に設定されたチャネルプログラム3 1 4 に含まれるシリンド、トラック等の情報と入出力マップテーブル1 1 6 に基づくファイル名情報とからデータの転送元と転送先を特定することができる。そして、ゲスト 計算機ドライバ1 1 2 は、入出力命令処理を正しく 実行完了したことをGTERM にセットして割り込み処理を終了する。

【0081】ゲスト 計算機3 0 0 においては、入出力命令を処理するエミュレーションプログラム3 1 2 がホスト 計算機1 0 0 による入出力処理が完了するのをGTERM の終了コードセットをポーリングにより待っているが、ホスト 計算機1 0 0 の入出力命令処理が終了し、GTERM に処理結果がセットされたことを検出することで入出力命令の終了処理に進む。エミュレーションプログラム3 1 2 は、GTERM にセットされたホスト 計算機1 0 0 による入出力命令処理結果に基づいて入出力命令の実行終了条件コードをゲスト プロセッサ3 2 0 の内部にセットすると共にGTERM 領域をクリアしてエミュレーションプログラム実行状態を終了する。そして、ゲスト プロセッサ3 2 0 中の実行ユニット3 2 2 は、入出力命令の実行を終了して、次の機械語命令の実行に進む。

【0082】以上のようにして、本実施の形態においては、ゲスト 計算機3 0 0 からホスト I / O 装置1 5 0 への入出力要求に対する処理を行うことができるので、ゲスト 計算機3 0 0 は、入出力装置を持たなくてもよくなり、システムとしてのコストの削減を図ることができる。

【0083】本実施の形態によれば、入出力マップテーブル1 1 6 を設けたので、ゲスト 計算機3 0 0 が管理するI / O 装置アドレスと個々の入出力装置をホスト OS 1 1 1 によって管理される1 つのファイルとを対応づけることができ、ゲスト 計算機3 0 0 の転送相手となる入出力装置又はファイルを特定できるようにした。その一方で、ゲスト 計算機3 0 0 が管理する仮想的な固定ディスク装置の物理位置をホスト 計算機1 0 0 に渡すことで、ホスト 計算機1 0 0 の転送相手を特定できるようにした。つまり、データの転送元と転送先が特定できるので、ゲスト 計算機3 0 0 は、ホスト 計算機1 0 0 におけるホスト I / O 装置1 5 0 を共用することができるようになる。

【0084】また、以上のようにしてゲスト 計算機3 0 0 が管理する仮想的な固定ディスク装置の物理位置とホ

スト OS 1 1 1 によって管理される1 つのファイルとを対応付けるようにしたので、I / O 装置制御プログラム1 1 3 は、ホスト 計算機1 0 0 が提供するファイルシステムをそのまま利用することができる。つまり、I / O 装置制御プログラム1 1 3 が通常のファイルアクセスを行えば、ゲスト 計算機ドライバ1 1 2 は、そのアクセスに応じてゲスト 計算機3 0 0 が管理する仮想的な固定ディスク装置の物理位置に変換しアクセスすることになる。これにより、ホスト 計算機1 0 0 側にとってみれば、ゲスト 計算機3 0 0 の仮想的な固定ディスク装置に対しても特別な作業なしでホスト OS 1 1 1 がファイルシステムに対して持つ機能、例えばディスクキャッシュによるゲスト 計算機3 0 0 から見た固定ディスクアクセス性能の向上、障害回復機能、セキュリティ機能などをそのまま適用することができる。

【0085】なお、ここではゲスト 計算機3 0 0 の実行する入出力命令処理の対象となる入出力装置としてHDD1 5 1 を例として動作説明をしたが、他のホスト I / O 装置1 5 0 中の入出力装置、例えばMT 1 5 2 やLAN 1 5 3 についても同様の動作によりゲスト 計算機3 0 0 からの入出力命令処理を実行することができる。

【0086】ゲスト 計算機3 0 0 からの入出力命令処理で指定されたI / O 装置の種類によってホスト 計算機1 0 0 側で処理が異なる部分は、ゲスト 計算機ドライバ1 1 2 の処理中で以下の2 点である。

【0087】第1 に入出力命令で指定されたI / O 装置アドレスに対応するホスト I / O 装置にマッピングする時に使う入出力マップテーブル1 1 6 のエントリが異なることである。第2 にI / O マップ機構により特定されたホスト I / O 装置1 5 0 の種類に対応したI / O 装置制御プログラム1 1 3 に対して入出力動作の開始を指示することである。

【0088】また、I / O 装置制御プログラム1 1 3 を各入出力装置に対して個々に設けたので、プリエンプティブ・マルチタスク処理機能を有するオペレーティングシステムの下では、各入出力装置を同時並行して動作させることができる。また、1 台の入出力装置に対して複数の処理を同時並行して動作させることができる。

【0089】(2) ゲスト 計算機3 0 0 の入出力命令で指定されたデータ転送動作
上記1 . の入出力命令実行動作では、ゲスト 計算機3 0 0 がホスト I / O 装置1 5 0 に格納されたファイルへのアクセスを行うための割り込み制御等の制御の流れについて説明した。ここでは、データ転送を実際に行うための動作について同じく図1 1 及び図1 2 に基づいて説明する。

【0090】前述したように、入出力処理の動作の開始を指示されたHDD1 5 1 に対するI / O 装置制御プログラム1 1 3 a は、GDATA 情報を基にゲスト 主記憶装置3 1 0 上にあるチャネルプログラム3 1 4 を読みだ

10

20

30

40

50

し、これを解釈する。

【0091】ここで、ゲスト 計算機300における仮想的な固定ディスク装置に対して出されたチャネルプログラム314中のコマンドを、データを実際に格納するホストI/O装置150中のHDD151への対応付ける操作について図10を用いて説明する。

【0092】図10において、まず、ゲスト 計算機300において実行される入出力命令の機械語中に指定されるI/O装置アドレスは、ゲスト 計算機ドライバ112において入出力マップテーブル116によりホストOS 111が管理するファイルシステム中の特定のファイルに対応付けられる。続いて、I/O装置制御プログラム113aは、ゲスト 主記憶装置310中に格納されたチャネルプログラム314を読み出す。そして、I/O装置制御プログラム113aは、チャネルプログラム314中のチャネルコマンドに付随するゲスト 計算機300での固定ディスク装置内の物理位置情報、例えばシリンダx、トラックy、セクタzといった値を、ホストOS 111が管理するファイルシステム中の特定のファイル中の論理的な位置情報、例えばレコード番号等に1対1に対応付ける。つまり、ゲスト 計算機300における1本の固定ディスク装置は、ホスト 計算機100上ではホストOS 111が管理する1つのファイルに対応付けられることになる。そして、HDD151に対するI/O装置制御プログラム113aは、ホストOS 111に対して特定のファイル中の論理的な位置情報を指定してファイルアクセス要求を出すと、ホストOS 111は自身で管理するファイルシステム情報より実際のHDD151上の物理位置を得て必要なHDD151上のデータをアクセスし、その結果をI/O装置制御プログラム113aに返す。このように、ゲスト 計算機300において固定ディスク装置に対する入出力処理を、ホスト 計算機100のHDD151にマッピングを行うことが可能になる。

【0093】I/O装置制御プログラム113aは、上記した通りホスト 計算機100のHDD151に格納されたファイルとゲスト 計算機300における仮想的な固定ディスク装置との間でデータ転送をシステムバス130を介して行う。そして、ゲストOS 311は、仮想的な固定ディスク装置の物理位置とゲスト 主記憶装置310とを対応づけるので、ゲスト 計算機300は、HDD151に対するファイルアクセスを実際に行うことができたということになる。

【0094】ここで、ホストI/O装置150とゲスト 主記憶装置310間でのデータ転送において必要となることは、ホスト 計算機100とゲスト 計算機300とがそれぞれ処理するデータの並び順(バイトオーダ)を予め知っておき、ゲスト 計算機300において適合するデータの並び順をゲストOS 311あるいはゲスト 計算機300のアプリケーションプログラム313から見て保

証することである。この目的のために、バイトオーダ変換機構332を設け、ホスト 計算機100とゲスト 計算機300との間の双方向のデータ転送を行うことができるようにした。本実施の形態においては、バイトオーダ変換機構332をゲスト 制御装置330に設けたが、ホスト 制御装置140に設けてもよい。

【0095】このバイトオーダ変換機構332の動作について図9を用いて説明する。

【0096】図9において、バイトオーダ変換指示信号は、予めシステムバスI/F制御部331においてシステム立上げ時にバイトオーダを変換する若しくはしないのどちらか一方の値に固定的にセットされる。もし、バイトオーダを変換しないようにセットされた場合には、マルチプレクサ336、337においてバイト位置の変換は行われず、システムバス130とゲスト 主記憶装置310との間のデータ転送は、同じバイト位置のまま行われる。バイトオーダ変換指示信号がバイトオーダを変換するようにセットされた場合には、マルチプレクサ336、337においてシステムバス130とゲスト 主記憶装置310間でのデータ転送をする場合に両方向共に最上位バイトと最下位バイト、最上位バイト側から数えて2番目のバイトと最下位バイトから数えて2番目のバイト、・・・というようにバイトデータの変換を行う。そして、システムバス130からゲスト 主記憶装置310に対してデータ転送が行われる時には、システムバスI/F制御部331から供給される図中のゲスト 主記憶バス出力可信号が有意となり、マルチプレクサ336で選択出力されたデータをバッファ338を通してゲスト 主記憶バス340上に出力する。また逆に、ゲスト 主記憶装置310からシステムバス130に対してデータ転送が行われる時には、システムバスI/F制御部331から供給される図中のシステムバス出力可信号が有意となり、マルチプレクサ337で選択出力されたデータをバッファ339を通してシステムバス130上に出力する。

【0097】以上のように、バイトオーダ変換機構を動作させることで、ゲスト 計算機300におけるデータ処理時におけるバイトデータの並び順とホスト 計算機100のデータ処理のバイトデータの並び順とが異なるものであったとしても互換性を保持することが可能になる。すなわち、ホスト 計算機100とゲスト 計算機300が異なる機種であっても、本実施の形態におけるバイトオーダ変換機構332を用いることで、双方向のデータ転送を行うことができるようになる。

【0098】次に、磁気テープ装置MT152のようにデータの記録媒体が交換可能であるようなI/O装置については、装着されている記録媒体がホストOS 111あるいはゲストOS 311のどちらで取り扱われる形式になっているかにより交換媒体を装着するI/O装置を排他利用する。つまり、ホストOS 111が取り扱う媒

体形式の磁気テープが装着されている場合には、ホスト OS 1 1 1 からの入出力処理を受け付けて実行する。一方、ゲスト OS 3 1 1 が取り扱う媒体形式の磁気テープが装着されている場合には、ゲスト 計算機 3 0 0 から発行される入出力処理をホスト 計算機 1 0 0 の I / O 装置制御プログラム 1 1 3 b がホスト OS 1 1 1 を介して実行する。なお、ホスト OS 1 1 1 中の MT 1 5 2 の動作制御を行う デバイスドライバは、ホスト OS 1 1 1 及びゲスト OS 3 1 1 両方の媒体形式に対する動作制御機能をサポートするように構成されている。

【 0 0 9 9 】 仮に、MT 1 5 2 にホスト OS 1 1 1 がサポートする形式の媒体が装着されているときにゲスト 計算機 3 0 0 から MT 1 5 2 に対する入出力命令が実行された場合には、ホスト 計算機 1 0 0 の MT 1 5 2 に対応する I / O 装置制御プログラム 1 1 3 b からホスト OS 1 1 1 を介して入出力アクセスが発行されるが、装着されている媒体形式がゲスト OS 3 1 1 が取り扱う形式とは異なることを認識して入出力処理を行わない。逆に、ゲスト OS 3 1 1 が扱う形式の媒体を MT 1 5 2 に装着しているときにホスト 計算機 1 0 0 のアプリケーションプログラム 3 1 3 から MT 1 5 2 に対するアクセスが出された場合にも同様に媒体形式の違いによりホスト 計算機 1 0 0 のアプリケーションプログラム 1 1 5 からの入出力処理は行われない。

【 0 1 0 0 】 更に、ホスト 計算機 1 0 0 とゲスト 計算機 3 0 0 との間で扱う文字データの表記形式即ち文字コードが異なっている場合でも、ホスト I / O 装置 1 5 0 に格納されたゲスト 計算機 3 0 0 が取り扱うデータをホスト 計算機 1 0 0 は何等意識することなくシステム処理が可能である。

【 0 1 0 1 】 チャネルプログラム 3 1 4 中の 1 つのチャネルコマンドに対する入出力動作が終了すると、I / O 装置制御プログラム 1 1 3 a は、引き続いて次のチャネルコマンドを読みだして入出力動作を続行する。このようにして、ゲスト 計算機 3 0 0 の入出力命令に基づいて起動された所定の I / O 装置制御プログラム 1 1 3 a は、チャネルプログラム 3 1 4 において終了が指定されるまでチャネルコマンドを解釈しては実行することを自律的に行う。

【 0 1 0 2 】 つまり、本実施の形態によれば、ゲスト 計算機 3 0 0 の入出力命令であってもホスト 計算機 1 0 0 側の I / O 装置制御プログラム 1 1 3 の実行によりデータ転送が行われることになるので、ゲスト プロセッサ 3 2 0 は、ゲスト 計算機 3 0 0 の入出力要求に基づくデータ転送のために何ら使用されることはない。従って、ゲスト 計算機 3 0 0 にしてみれば、このデータ転送に関する入出力処理と他の処理とを見かけ上同時並行して行っているようになるので、ゲスト 計算機 3 0 0 におけるシステム性能の向上を図ることができる。

【 0 1 0 3 】 また、ホスト I / O 装置 1 5 0 に対する全

てのアクセスは、ホスト OS 1 1 1 から一括して動作制御を受けるように構成することで、ゲスト 計算機 3 0 0 からの入出力アクセスとホスト 計算機からの入出力アクセスとの排他制御などをホスト OS 1 1 1 の機能により実現することができる。これにより、柔軟な複合計算機システムが容易に構築することができる。

【 0 1 0 4 】 (3) ゲスト 計算機 3 0 0 の入出力終了割り込み処理動作

上記 2 . の入出力データ転送動作では、ゲスト 計算機 3 0 0 によるホスト I / O 装置 1 5 0 に格納されたファイルへのアクセス処理をホスト 計算機 1 0 0 側の動作により行う処理の流れについて説明した。図 1 3 . 及び図 1 4 は、ゲスト 計算機 3 0 0 による入出力処理の完了時において発生する入出力終了割り込み処理を本実施の形態の複合計算機システムにおいて実行する際の内部処理フローを示した図であり、図 1 4 は図 1 3 の続きの図であるが、ここでは、これらの図に基づきデータ転送終了後における割り込み処理の動作について説明する。

【 0 1 0 5 】 I / O 装置制御プログラム 1 1 3 a は、チャネルプログラム 3 1 4 の中に終了を意味するチャネルコマンドを検出し入出力処理を終了すると、ゲスト 計算機ドライバ 1 1 2 に対して入出力終了割り込み要求を通知する。そして、I / O 装置制御プログラム 1 1 3 a は、ゲスト 計算機 3 0 0 から自分の I / O 装置アドレスに対する入出力終了割り込みが受け付けられるまで待ちつづける。

【 0 1 0 6 】 一方、この入出力終了割り込み要求を受け取ったゲスト 計算機ドライバ 1 1 2 は、ゲスト 計算機 3 0 0 に対する HDD 1 5 1 からの割り込み要求をドライバ内部状態としてペンディングすると共に、ゲスト 制御装置 3 3 0 中にある割り込み受信レジスタ 3 3 5 にシステムバス 1 3 0 を介して入出力終了割り込み要求をセットする。この入出力終了割り込み要求の場合には I O I ビットがセットされる。

【 0 1 0 7 】 割り込み受信レジスタ 3 3 5 の I O I ビットがセットされると、ゲスト プロセッサ 3 2 0 に対して割り込み要求が通知される。そして、ゲスト プロセッサ 3 2 0 は、割り込み受付可能状態になると、I O I ビットによる割り込み要求が受け付けられて、ゲスト プロセッサ 3 2 0 は割り込み処理に入る。

【 0 1 0 8 】 割り込み処理に入ったゲスト プロセッサ 3 2 0 は、割り込み受信レジスタ 3 3 5 を読み取り、I O I ビットがセットされていることから割り込み要因を入出力終了割り込みであると認識する。なお、割り込み受信レジスタ 3 3 5 は、ゲスト プロセッサ 3 2 0 からのレジスタ読み取り後にクリアされる。

【 0 1 0 9 】 入出力終了割り込みを認識したゲスト プロセッサ 3 2 0 は、実行ユニット 3 2 2 において入出力終了割り込み処理を実行するエミュレーションプログラム 3 1 2 を呼び出す。

【 0 1 1 0 】呼び出されたエミュレーションプログラム 3 1 2 は、入出力終了割込みを受け付けたことを計算機間通信記憶領域 3 1 5 内の G F A C T O R にセットし、引き続いてエミュレーションプログラム 3 1 2 からホスト計算機 1 0 0 に対する割込み要求としてゲストステータスレジスタ 3 3 4 中の E M I ビットにセットする。そして、エミュレーションプログラム 3 1 2 は、ホスト計算機 1 0 0 による入出力終了割込み処理が完了するのを計算機間通信記憶領域 3 1 5 中の G T E R M をポーリングすることで待つ。

【 0 1 1 1 】ゲストステータスレジスタ 3 3 4 に E M I ビットがセットされると、割込み要求レジスタ 3 3 3 がセットされて、システムバス 1 3 0 及びホスト制御装置 1 4 0 経由でホストプロセッサ 1 2 0 に対してゲスト計算機 3 0 0 からの割込み要求が入る。ホストプロセッサ 1 2 0 は、要求された割込みの受け付け可能タイミングになると、このゲスト計算機 3 0 0 からの割込み要求を受け付け割込み処理に入るためにゲスト計算機ドライバ 1 1 2 を呼び出す。ゲスト計算機ドライバ 1 1 2 は、割込み要因を知るためにゲストステータスレジスタ 3 3 4 をシステムバス 1 3 0 経由で読みだす。このとき、ゲストステータスレジスタ 3 3 4 中の E M I ビット及び割込み要求レジスタ 3 3 3 はクリアされる。

【 0 1 1 2 】ゲストステータスレジスタ 3 3 4 を読みだしたゲスト計算機ドライバ 1 1 2 は、E M I ビットがセットされていることからエミュレーションプログラム 3 1 2 による割込み処理要求であることを認識する。続いて更に詳細な割込み要因を知るために計算機間通信記憶領域 3 1 5 内にある G F A C T O R を読みだす。読みだされた G F A C T O R には、エミュレーションプログラム 3 1 2 により入出力終了割込みの受け付けであることがセットされているので、ゲスト計算機ドライバ 1 1 2 は、入出力終了割込み受け付け処理を開始する。なお、このとき、前述した通りゲスト計算機ドライバ 1 1 2 の内部状態として H D D 1 5 1 からの入出力終了割込み要求をペンディングしている。

【 0 1 1 3 】ところで、前述した H D D 1 5 1 に対する入出力動作を制御する I / O 装置制御プログラム 1 1 3 a が入出力終了割込み要求を行ってから、ゲスト計算機 3 0 0 中のエミュレーションプログラム 3 1 2 による入出力終了割込み受け付け処理がホスト計算機 1 0 0 に届く間に、H D D 1 5 1 以外のホスト I / O 装置、例えば M T 1 5 2 に対するゲスト計算機 3 0 0 からの入出力動作が I / O 装置制御プログラム 1 1 3 b により並行して行われて入出力処理が完了する場合もあり得る。このような場合にはゲスト計算機ドライバ 1 1 2 の内部に M T 1 5 2 に対する I / O 装置制御プログラム 1 1 3 b からの入出力終了割込み要求が H D D 1 5 1 からの入出力終了割込み要求に加えてペンディングされていることになる。そこで、ゲスト計算機ドライバ 1 1 2 は、ペンディ

ングされている入出力終了割込み要求の中から優先順位の最も高い 1 つのホスト I / O 装置 1 5 0、例えば H D D 1 5 1 からの要求を選択して、対応する I / O 装置制御プログラム 1 1 3 a に対して入出力終了割込みの受け付けを通知すると共に、ゲスト計算機 3 0 0 からの入出力終了割込み受け付け要因に対する割込み処理を終了する。

【 0 1 1 4 】入出力終了割込みの受け付けを受け取った I / O 装置制御プログラム 1 1 3 (この例では 1 1 3 a) は、ゲスト計算機 3 0 0 の入出力終了割込みに必要な入出力動作終了ステータスなどの各種情報をゲスト主記憶装置 3 1 0 の所定のアドレスに格納する。この格納処理が完了すると、I / O 装置制御プログラム 1 1 3 は、入出力終了割込みとしてホスト計算機 1 0 0 側での処理の完了をゲスト計算機 3 0 0 のエミュレーションプログラム 3 1 2 に対して通知するために、計算機間通信記憶領域 3 1 5 中の G T E R M に終了データをセットする。そして、I / O 装置制御プログラム 1 1 3 における入出力終了割込み処理は完了する。

【 0 1 1 5 】ホスト計算機 1 0 0 側による入出力終了割込み処理が終了したことを示すデータが G T E R M にセットされると、これをポーリングしていたエミュレーションプログラム 3 1 2 は、ホスト計算機 1 0 0 による入出力終了割込み受け付け処理が完了したことを認識して、ゲスト計算機 3 0 0 による入出力終了割込み処理を起動する。この時にエミュレーションプログラム 3 1 2 は、G T E R M の内容をクリアする。そして、エミュレーションプログラム 3 1 2 は、ゲストプロセッサ 3 2 0 内にあるプログラムカウンタに入出力終了割込み処理ルーチンの先頭アドレスをセットして、エミュレーションプログラム実行状態から抜け出す。この操作により、ゲストプロセッサ 3 2 0 において入出力終了割込み処理のゲスト OS 3 1 1 ルーチンの先頭アドレスから機械語命令の実行が開始されることになる。

【 0 1 1 6 】以上のようにして、ゲスト計算機 3 0 0 からのホスト I / O 装置 1 5 0 へのアクセス要求に対するデータ転送後の終了処理が行われるが、本実施の形態によれば、ゲスト計算機 3 0 0、ホスト計算機 1 0 0 及びホスト I / O 装置 1 5 0 の間はホスト計算機 1 0 0 のシステムバス 1 3 0 により結合されているために、従来ネットワーク経由で通信していたシステムと比較してより高性能なシステムを構築することができる。

【 0 1 1 7 】また、ホスト計算機 1 0 0 とゲスト計算機 3 0 0 とで異なるアーキテクチャとして構成することができるため、低コストの I / O 装置等を使うことのできるアーキテクチャのシステムをホスト計算機として選択することで、より低コストの複合計算機システムを提供することができる。

【 0 1 1 8 】(4) ゲスト計算機 3 0 0 の専用 I / O 装置 3 6 0 への入出力命令実行動作

次に、ゲスト計算機 3 0 0 からゲスト専用 I / O 装置 3

10

20

30

40

50

6 0 に対する入出力命令が実行されたときの動作について説明する。これは、上記1 . に記載したゲスト 計算機 3 0 0 の入出力命令実行動作において、実行ユニット 3 2 2 は、入出力命令に含まれるI / O 装置アドレス情報と入出力識別テーブル3 2 3 の識別情報とに基づいて、命令デコード3 2 1 から受け取った入出力命令を、ゲスト 専用I / O 装置3 6 0 に対する命令と判断した場合の処理である。この判断は、図5 に示した識別情報、すなわちI / O 装置アドレスとI / O 装置アドレスに対応した入出力装置を参照すれば容易に行うことができる。

【 0 1 1 9 】 実行ユニット 3 2 2 は、ゲスト 専用I / O 装置3 6 0 に対する命令と判断すると、独自のI / O バス3 5 0 経由でゲスト 専用I / O 装置3 6 0 のうち指定された入出力装置に対する入出力処理を開始するためのコマンドを発行する。これ以降、独自のI / O バス3 5 0 経由で行われる入出力処理は、従来と同様の処理手順でアクセスを行うため詳細な説明は省略する。

【 0 1 2 0 】 本実施の形態においては、以上のように、ゲスト 計算機3 0 0 は、入出力識別テーブル3 2 3 の識別情報等に基づいて転送相手を識別し 特定することができる。前述したように、本実施の形態においては、ホスト I / O 装置1 5 0 を共用することができるが、ゲスト 計算機3 0 0 においてアクセス頻度の高い入出力装置をゲスト 専用I / O 装置3 6 0 とすることで、ホスト プロセッサ1 2 0 にかかる負荷を軽減し、システム全体として高性能な複合計算機システムを提供することができる。

【 0 1 2 1 】 また、特殊な入出力装置をゲスト 専用I / O 装置3 6 0 とし、固定ディスク 装置等汎用的な入出力装置のみをホスト I / O 装置1 5 0 として共用することで、エミュレーションプログラム3 1 2 の作成を容易にし、その負荷が軽減した分ゲスト プロセッサ3 2 0 に多種に渡る入出力装置をサポート させるようにすることができる。

【 0 1 2 2 】 (5) 操作員とゲスト 計算機3 0 0 とのコミュニケーション動作

上記1 ~ 3 で説明したことから、本実施の形態によれば、ホスト I / O 装置1 5 0 をホスト 計算機1 0 0 及びゲスト 計算機3 0 0 で共用することができ、ゲスト 計算機3 0 0 からの入出力要求に基づく データ転送処理をゲスト プロセッサ3 2 0 を用いることなく実現することができることがわかった。また、上記4 により、ゲスト 計算機3 0 0 は、共用の入出力装置と 専用の入出力装置とを使い分けることができることがわかった。

【 0 1 2 3 】 更に、本実施の形態においては、ホスト 計算機1 0 0 側に接続された端末1 5 4 からゲスト 計算機3 0 0 のゲスト 主記憶装置3 1 0 やゲスト 制御装置3 3 0 の設定内容を参照することができることも特徴の1 つとしている。

【 0 1 2 4 】 図1 5 及び図1 6 は、本実施の形態にお

る複合計算機システムにおいて、操作員とゲスト 計算機 3 0 0 とのインタフェース処理を行う 時の内部処理フローを示した図であり、図1 6 は図1 5 の続きの図であるが、次に、複合計算機システムの操作員とゲスト 計算機 3 0 0 との間で必要なコミュニケーション動作について図1 5 及び図1 6 に基づいて説明する。

【 0 1 2 5 】 ゲスト 計算機3 0 0 に対する操作員から指示される操作の例としては、ゲスト 計算機3 0 0 の初期化、ゲスト 計算機3 0 0 の命令実行開始や停止、ゲスト 主記憶装置3 1 0 の内容表示や内容変更、ゲスト プロセッサ3 2 0 内のレジスタ内容表示や内容変更などがある。ここでは、ゲスト プロセッサ3 2 0 によるゲスト 主記憶装置3 1 0 の内容表示動作を例にあげて、操作員とゲスト 計算機3 0 0 とのコミュニケーション動作について説明する。

【 0 1 2 6 】 まず、操作員がホスト I / O 装置1 5 0 の内、操作員インタフェースを分担する入出力装置である端末1 5 4 を介してゲスト 主記憶装置3 1 0 の指定したアドレスの内容表示をホスト 計算機1 0 0 に対して指示する。すると、ホスト OS 1 1 1 の上で動作しているゲスト 計算機3 0 0 に対する操作員インタフェース制御プログラム1 1 4 が、ホスト OS 1 1 1 経由で上記の操作員からの指示を受け取る。すると、操作員インタフェース制御プログラム1 1 4 は、操作員からの指示内容を解釈してゲスト 主記憶装置3 1 0 の内容を読み出して端末1 5 4 上に結果を表示することを認識する。そして、操作員インタフェース制御プログラム1 1 4 は、ゲスト 計算機3 0 0 に対してゲスト 主記憶装置3 1 0 の内容を読み出す処理を要求するために、計算機間通信記憶領域3 1 5 中のホスト 計算機1 0 0 からゲスト 計算機3 0 0 に対する割込み要因としてH F A C T O R にゲスト 主記憶装置3 1 0 の読み出し処理に対応するコードをセットする。更に、操作員インタフェース制御プログラム1 1 4 は、計算機間通信記憶領域3 1 5 中のH D A T A 1 に読み出すべきゲスト 主記憶装置3 1 0 内のアドレス情報をセットする。読み出すべきデータは、前述したようにゲスト 主記憶装置3 1 0 等の内容であり、その格納場所つまりそのデータの格納アドレスは既知である。この格納アドレスをアドレス情報としてセットする。そして、ゲスト 計算機ドライバ1 1 2 及びシステムバス1 3 0 を介してゲスト 制御装置3 3 0 中にある割込み受信レジスタ3 3 5 のO P I ビットをセットすることでゲスト 計算機3 0 0 に対する操作員インタフェース処理割込み要求を伝える。操作員インタフェース制御プログラム1 1 4 は、ゲスト 計算機3 0 0 による操作員インタフェース割込み処理が終了するのを計算機間通信記憶領域3 1 5 中のH T E R M の内容をポーリングして待つ。

【 0 1 2 7 】 割込み受信レジスタ3 3 5 のO P I ビットがセットされると、ゲスト 制御装置3 3 0 は、ゲスト プロセッサ3 2 0 に対して割込み要求を行う。そして、ゲ

ストプロセッサ320は、割込み要求を受け付け可能状態になると、割込み処理を開始する。割込み処理を開始したゲストプロセッサ320は、まずゲスト制御装置330から割込み受信レジスタ335を読み出す。この読み出し操作により、割込み受信レジスタ335はクリアされる。読み出した内容に含まれるOPIビットはセットされているので、ゲストプロセッサ320は、操作員インタフェース処理割込み要求であることを認識する。ここで、ゲストプロセッサ320の実行ユニット322は、エミュレーション実行状態に入り、操作員インタフェース割込み処理を実行するエミュレーションプログラム312を起動するという所定の処理を行う。

【0128】操作員インタフェース割込み処理を行うエミュレーションプログラム312は、計算機間通信記憶領域315中のHFACTORを読み出して、操作員インタフェース割込みの詳細要因情報からゲスト主記憶装置310に対するホスト計算機100からのデータ読み出しであることを認識する。次に、エミュレーションプログラム312は、計算機間通信記憶領域315中のHDATA1の内容を読み出して、ホスト計算機100が指定しているゲスト主記憶装置310中のアドレスを知る。そして、エミュレーションプログラム312は、指定されたゲスト主記憶装置310のアドレスを用いてHFACTORで指定されたアクセス情報によりゲスト主記憶装置310からデータを読み出し、結果をHDATA2にセットする。最後に、エミュレーションプログラム312は、終了コードをHTERMにセットした後、操作員インタフェース割込み処理を終了してエミュレーション実行状態から脱する。

【0129】HTERMに処理終了を示すデータがセットされると、HTERMをポーリングしていた操作員インタフェース制御プログラム114は、ゲスト計算機300における操作員インタフェース割込み処理が終了したことを検出する。そして、操作員インタフェース制御プログラム114は、HTERM、及びHDATA2の内容を読み出してエミュレーションプログラム312によりセットされたゲスト計算機300における操作員インタフェース割込み処理の終了状態をチェックした後にHTERMをクリアする。次に、操作員インタフェース制御プログラム114は、読み出したHDATA2の値をホストI/O装置150、特にここでは端末154に対してホストOS111経由で出力することで操作員が指示したゲスト主記憶装置310のアドレスに対する内容を表示する。そして、操作員インタフェース制御プログラム114は、操作員が指示したゲスト計算機300とのコミュニケーション処理を完了する。

【0130】以上のように、本実施の形態によれば、ホスト計算機100側に接続された端末154からゲスト計算機300に設定された内容を参照することができるので、ゲスト計算機300側に別途操作員が使用する端

末を接続する必要がない。すなわち、ゲスト計算機300において、操作員インタフェース動作機能をホスト計算機の端末(操作員インタフェース入出力装置)154を共用して実現することができることから、ゲスト計算機300専用の操作員インタフェース入出力装置が不要となりシステムを低コストで実現することができる。

【0131】(6)まとめ

これまで本実施の形態における複合計算機システムの主要な動作である、ゲスト計算機300の入出力命令実行動作、ゲスト計算機300の入出力命令で指定されたデータ転送動作、ゲスト計算機300の入出力終了割込み処理動作、ゲスト計算機300の専用I/O装置360への入出力命令実行動作及び操作員とゲスト計算機とのコミュニケーション動作について各々説明を行った。上記の5種類の動作のうちゲスト計算機300の専用I/O装置360への入出力命令実行動作以外は、いずれもホスト計算機100側におけるI/O装置制御プログラム113若しくは操作員インタフェース制御プログラム114と、ゲスト計算機300側におけるエミュレーションプログラム312とが互いに通信を行いながら実行される。ここで、ホストOS111は、プリエンティブ・マルチタスク処理機能を有することから、I/O装置制御プログラム113、操作員インタフェース制御プログラム114及びアプリケーションプログラム115は、いずれもホストOS111のタスクスイッチ周期に従って強制的に実行が切り換えられることになる。すなわち、ホスト計算機100においてゲスト計算機300の入出力命令処理や入出力データ操作や入出力終了割込み処理や操作員からのゲスト計算機300に対するコミュニケーション処理はタイムシェアリングされて実行されるため、ゲスト計算機300や操作員からは見掛け上リアルタイムに処理が行われていることになる。更に、ホスト計算機100においては、アプリケーションプログラム115も同様にタイムシェアリングされて実行されることから、例えばホスト計算機100をサーバ、ゲスト計算機300をクライアントとする分散コンピュータシステムを構築する場合には、クライアントからサーバに対して発行されるサービス要求をホスト計算機100においてサーバアプリケーションを動作させることで処理するといったクライアント/サーバシステムを構成することができる。

【0132】ところで、本実施の形態において、ホスト計算機とは、複数の計算機で構築されたシステムにおいてホストとなりうる1乃至複数の計算機のことをいい、ゲスト計算機とは、ホスト計算機でない計算機のことをいい、特に断らない限り一般的な独立した計算機を構築する。従って、上記とは逆に、ホスト計算機100をクライアント、ゲスト計算機300をサーバとする分散コンピュータシステム構成とした場合にも、ホスト計算機100上で動作するクライアント・アプリケーションプ

ログラム115からサーバであるゲスト 計算機300に対して発行されるサービス要求をゲスト 計算機300上のアプリケーションプログラム313によって処理するといったクライアント /サーバシステムを構成することも可能である。

【0133】特に、本実施の形態によれば、ホスト 計算機100とゲスト 計算機300とをシステムバス130で接続することで、クライアント /サーバを一体として実現することができ、完全に独立した2台のコンピュータシステムで構築したクライアント /サーバシステムと比べると低コストで同じシステムが構築できる。更に、LAN等により接続して構築したシステムより高速なシステムを提供することができる。

【0134】一方、ゲスト 計算機300に独立したI /O装置や操作員インタフェース入出力装置が接続されて、ゲスト 計算機300だけで完全なコンピュータシステムを構成する場合であっても、上記実施の形態の動作説明で述べてきた通り、ゲスト 計算機300において実行される入出力命令処理、入出力終了割り込み、操作員インタフェース割り込み処理は、全てゲスト 主記憶装置310内の特定領域に配置されて、ゲスト プロセッサ320の機械語命令セットを用いたプログラムにより実行されるエミュレーションプログラム312により実現されている。従って、本実施の形態によれば、I /O装置構成や操作員インタフェース入出力装置の構成が変わったことによる処理部分を改訂したエミュレーションプログラム312に交換することだけで、何等ゲスト プロセッサ320の機械語命令セットとハードウェア、及びゲスト OS311、チャネルプログラム314、アプリケーションプログラム313を変更することなしに前述したゲスト 計算機300と同じ機能を実現することができる。

【0135】更に、これまでの動作説明から、本実施の形態におけるホスト 計算機100及びゲスト 計算機300の間、更にホスト プロセッサ120及びゲスト プロセッサ320の間では、システムバス130に接続するという部分を除いて互いにアーキテクチャとしての依存関係がないことは自明である。

【0136】本実施の形態では、ホスト プロセッサ120及びゲスト プロセッサ320共に1つのシングルプロセッサ構成による複合計算機システムの動作について示したが、ホスト プロセッサ120が複数個、あるいはゲスト プロセッサ320が複数個から構成されるマルチプロセッサシステム構成についても、システムバス130に上記構成を有するホスト 計算機、ゲスト 計算機を接続し、必要な構成要素を重畳させて持たせることで容易に適用することができる。

【0137】また、本実施の形態では、図1及び図3に示したように、上記の「(4)ゲスト 計算機300の専用I /O装置360への入出力命令実行動作」を説明するために、ゲスト 計算機300に独自のI /Oバス35

0及びゲスト 専用I /O装置360を設けた構成としたが、これ以外の動作においては、これらの構成はなくてもよい。

【0138】複合機計算システムにおけるインストール /アンインストール方法

次に、本実施の形態におけるインストール及びアンインストールの方法について説明する。本実施の形態において特徴的なことは、単一のインストーラ及び単一のインストール媒体を利用してインストールできるようにしたことであり、すなわち、図1に示した構成においては、ゲスト 計算機300へのインストール /アンインストールをホスト 計算機100側からできるようにしたことである。

【0139】図17はインストール及びアンインストール作業を示す模式図である。この実施の形態においては、ホスト 計算機100のオペレーティングシステム111は、既にインストールされているものとする。図17において、フロッピーディスクやCD-ROMのようなインストール媒体200には、インストーラ211、アンインストーラ212の他に、ゲスト 計算機制御プログラムであるゲスト 計算機ドライバ112、I /O装置制御プログラム113、操作員インタフェース制御プログラム114、それ以外のホスト アプリケーションプログラム115及びゲスト OS311とゲスト アプリケーションプログラム313とを物理ディスクのイメージでそのままマッピングしたファイル210が含まれる。HDD151は、インストール先となる図3に示した複合計算機システムの共用の固定ディスクである。なお、矢印Aはインストール作業におけるファイルのコピーを表し、矢印Bはアンインストール作業におけるファイルの削除を意味する。インストール前は、ホスト OS111のみがHDD151に格納されており、インストール作業によって、ゲスト 計算機ドライバ112、I /O装置制御プログラム113、操作員インタフェース制御プログラム114、それ以外のホスト アプリケーションプログラム115及び物理的な固定ディスク装置のイメージそのままマッピングしたファイル210がHDD151にコピーされる。

【0140】ゲスト OS311から見ると、HDD151の中の一つのファイル210が、一つの固定ディスク装置として認識されることになる。一方、ホスト OS111からしてみると、ゲスト OS311とゲスト アプリケーションプログラム313とを一つのファイル210としてインストール媒体200に格納しておくことで、ホスト 計算機100のみならずゲスト 計算機300へのインストール作業を、ホスト OS111上で動作するインストーラ211のみで行うことができる。

【0141】インストール時には、HDD151に予め格納されていたホスト OS111を、ホスト 主記憶装置110にロードして動作させる。次に、インストーラ2

1 1 を同じくホスト 主記憶装置1 1 0 にロードして動作させる。インストーラ2 1 1 は、インストール媒体2 0 0 からホスト 計算機1 0 0 用のソフトウェアとして、ゲスト 計算機ドライバ1 1 2、I / O装置制御プログラム1 1 3、操作員インタフェース制御プログラム1 1 4、それ以外のホスト アプリケーションプログラム1 1 5 及びゲスト 計算機3 0 0 用のソフトウェアであるファイル2 1 0 を共用のHDD1 5 1 にコピーする。このように、ゲスト OS 3 1 1 をインストールする際に、ゲスト 計算機3 0 0 を全く動作させることなくゲスト OS 3 1 1 をインストールすることができる。ゲスト OS 3 1 1 が動作するために必要となる、複合計算機システムに接続されているホスト I / O装置1 5 0 の情報については、ゲスト OS 3 1 1 動作後に、ホスト OS 1 1 1 が制御しているI / O装置制御プログラム1 1 3とゲスト 計算機ドライバ1 1 2 を通じてゲスト OS 3 1 1 が認識する。複合計算機のホスト I / O装置1 5 0 は、すべてホスト OS 1 1 1 が制御しており、ホスト OS 1 1 1 が認識しているホスト I / O装置1 5 0 の情報に基づいてI / O装置制御プログラム1 1 3とゲスト 計算機ドライバ1 1 2 が動作するので、ゲスト OS 3 1 1 にI / O装置の情報を知らせることなく、ゲスト OS 3 1 1 をインストールできる。

【 0 1 4 2 】アンインストールは、複合計算機のHDD 1 5 1 から、ゲスト 計算機3 0 0 用の情報を全て削除することで行われる。ゲスト 計算機3 0 0 用の情報とは、ホスト 計算機1 0 0 上のアプリケーションプログラムであるゲスト 計算機ドライバ1 1 2、I / O装置制御プログラム1 1 3、操作員インタフェース制御プログラム1 1 4、ファイル2 1 0 及びゲスト 計算機3 0 0 用のシステム情報がある。インストーラ2 1 1 は、インストール時に単にファイルのコピーを行うだけではなく、ゲスト 計算機3 0 0 用のシステム情報をHDD1 5 1 の特定の領域に書き込む。ゲスト 計算機3 0 0 用のシステム情報は、図1 8 と図1 9 に示す様にゲスト 計算機制御プログラムの情報とホスト I / O装置1 5 0 の情報とから構成され、ゲスト 計算機制御プログラムは、ゲスト 計算機3 0 0 の動作中にシステム情報を参照しパス名等の制約条件に従って動作する。

【 0 1 4 3 】インストール媒体2 0 0 に格納されたアンインストール2 1 2 をホスト 主記憶装置1 1 0 にロードして動作させると、始めにゲスト 計算機3 0 0 用のシステム情報を参照する。ゲスト 計算機制御プログラムの情報からゲスト 計算機制御プログラムのパスを調べ、ゲスト 計算機制御プログラムを全て削除する。次に、ホスト I / O装置1 5 0 の情報からゲスト 計算機3 0 0 の固定ディスクに対応するファイルのパスを調べて全て削除する。ファイルの削除が終わったら、ゲスト 計算機3 0 0 用のシステム情報を全て削除する。

【 0 1 4 4 】このように複合計算機のアンインストール

2 1 2 は、アーキテクチャの異なるゲスト 計算機3 0 0 用ソフトウェアとホスト 計算機1 0 0 用ソフトウェアを一度の作業でかつ迅速にアンインストールできる。

【 0 1 4 5 】ところで、図2 0 は、共用のHDD1 5 1 に格納された上述のファイル2 1 0 の詳細を示した図であるが、ファイル2 1 0 は、インストール媒体2 0 0 からインストーラ2 1 1 によってHDD1 5 1 にコピーされたもので、前述した入出力マップテーブル1 1 6 によってゲスト 計算機3 0 0 のHDDに割り当てられるものである。このファイル2 1 0 は、ホスト OS 1 1 1 から見ると通常のファイルであり、先頭にホスト OS 1 1 1 用のヘッダ情報2 1 3 を持つ。次に、ホスト 計算機1 0 0 上のアプリケーションプログラムであるインストーラ2 1 1 およびI / O装置制御プログラム1 1 3 が、ファイル2 1 0 をゲスト 計算機3 0 0 用のファイルとして認識するためのインストーラ2 1 1 用のヘッダ情報2 1 4 を持つ。それ以後の物理的な領域が、ゲスト 計算機3 0 0 が入出力マップテーブル1 1 6 を通じてHDDとして認識する部分である。先頭の領域にはゲスト OS 3 1 1 が格納されており、それ以後の領域にゲスト アプリケーションプログラム3 1 3 が格納されている。このファイル2 1 0 は以下のように作成してインストール媒体2 0 0 に格納される。

【 0 1 4 6 】まず、ホスト 計算機1 0 0 上でゲスト 計算機3 0 0 のHDDで必要となるサイズの空ファイルを作成する。このことでホスト 計算機1 0 0 は、このファイルのOS 管理領域にホスト OS 用ヘッダ情報2 1 3 を書き込む。次にこの空ファイルのユーザ領域の先頭にインストーラ2 1 1 およびI / O装置制御プログラム1 1 3 用のインストーラ用のヘッダ情報2 1 4 を書き込む。この作業は、ホスト 計算機1 0 0 上でアプリケーションプログラム1 1 5 を用いて、直接空ファイルにアクセスして行うことができる。最後に、複合計算機ではなく単独で動作しているゲスト 計算機3 0 0 のHDD3 6 1 の内容を、物理イメージのまま、空ファイル内のインストーラ用のヘッダ情報2 1 4 の直後から1 バイトづつ書き込む。なお、HDD3 6 1 から読み込むデータは、ゲスト OS 3 1 1 とゲスト アプリケーションプログラム3 1 3 である。ゲスト 計算機3 0 0 のHDD3 6 1 の内容は、例えば次の様にしてコピーできる。まず、ホスト 計算機1 0 0 とゲスト 計算機3 0 0 とで物理的に共用できる（論理フォーマットは同じ必要はない）リムーバブル大規模記憶装置の媒体（例えばテープデバイス）に、ゲスト 計算機3 0 0 のHDD3 6 1 の内容を全て一時的にコピーする。次に、その大規模記憶装置の媒体をホスト 計算機1 0 0 に持ってきて、その内容をホスト 計算機1 0 0 上のアプリケーションプログラム1 1 5 を用いて1 バイトずつ物理的に読み込んで、1 バイトずつ空ファイルに書き込む。作成したファイルをインストール媒体2 0 0 にコピーする。このようにして、インストール媒体2

00を生成する。

【0147】図21は、インストーラ211又はアンインストーラ212と、ゲストOS311やゲスト計算機制御プログラムとしてのゲスト計算機ドライバ112、I/O装置制御プログラム113及び操作員インタフェース制御プログラム114との通信経路を示すものである。インストーラ211やアンインストーラ212が直接通信するのは、ゲスト計算機ドライバ112および操作員インタフェース制御プログラム114である。これらはすべてホストOS111上のアプリケーションプログラムであることから、ホストOS111を経由して通信することが可能である。ゲスト計算機ドライバ112は、計算機間通信記憶領域315やゲスト制御装置330を用いてゲストOS311と通信を行う。一方、操作員インタフェース制御プログラム114は、I/O装置制御プログラム113がホストI/O装置150の数(113a, 113b, 113c)だけあるので、すべてのI/O装置制御プログラム113と、ホストOS111を経由して通信を行う。

【0148】図22は、インストーラ211及びアンインストーラ212が図21の通信経路を用いて、システムを停止するフローチャートを示す。インストーラ211は、操作員インタフェース制御プログラム114と通信し、動作中かどうかを調べる監視手段と、動作中であれば操作員インタフェース制御プログラム114を停止するための停止手段を有する。操作員インタフェース制御プログラム114は、I/O装置制御プログラム113と通信し、I/O装置制御プログラム113が動作中かどうかを調べる監視手段と、動作中であればI/O装置制御プログラム113を停止するための停止手段を有する。また、I/O装置制御プログラム113は操作員インタフェース制御プログラム114と通信し、操作員インタフェース制御プログラム114が動作中かどうかを調べる監視手段を有し、I/O装置制御プログラム113起動時は、操作員インタフェース制御プログラム114が動作していることを前提とする。つまり、操作員インタフェース制御プログラム114が動作していないときは、I/O装置制御プログラム113は動作できないようにする。

【0149】このことで、まずインストーラ211は、操作員インタフェース制御プログラム114が動作中か調べ、動作していれば、操作員インタフェース制御プログラム114がI/O装置制御プログラム113が動作中かどうかを調べる。動作していれば停止手段を用いて、I/O装置制御プログラム113を停止する。次にインストーラが操作員インタフェース制御プログラム114を停止手段を用いて停止する。

【0150】同様に、インストーラ211は、ゲスト計算機ドライバ112と通信し、動作中かどうかを調べる監視手段と、動作中であればゲスト計算機ドライバ11

2を停止するための停止手段を有する。ゲスト計算機ドライバ112は、ゲストOS311と通信し、ゲストOS311が動作中かどうか調べる監視手段と、動作中であればゲストOS311を停止するための停止手段を有する。また、ゲストOS311はゲスト計算機ドライバ112と通信し、ゲスト計算機ドライバ112が動作中かどうか調べる監視手段を有し、ゲストOS311起動時は、ゲスト計算機ドライバ112が動作していることを前提とする。つまり、ゲスト計算機ドライバ112が動作していないときは、ゲストOS311は動作できないようにする。

【0151】このことで、まずインストーラ211は、ゲスト計算機ドライバ112が動作中か調べ、動作していれば、ゲスト計算機ドライバ112がゲストOS311が動作中かどうかを調べる。動作していれば停止手段を用いて、ゲストOS311を停止する。次にインストーラ211がゲスト計算機ドライバ112を停止手段を用いて停止する。これらの停止手段を用いることで、インストール(再インストールやバージョンアップ)時・アンインストール時に、ゲスト計算機制御プログラムやゲストOS311が動作中でもマニュアル操作によりそれらを停止しなくても、インストーラ又はアンインストール自身が自動的にそれらを停止してからインストールやアンインストールをすることができる。

【0152】

【発明の効果】本発明によれば、ホスト計算機のシステムバスに接続された入出力装置をゲスト計算機からも共用することが可能となり、かつ、ゲスト計算機が行う当該入出力装置に対する入出力処理をゲスト計算機のゲストプロセッサを用いることなく行うことが可能となる。

【0153】また、ゲスト計算機、ホスト計算機及び入出力装置の間はホスト計算機のシステムバスにより結合されているために、従来ネットワーク経由で通信していたシステムと比較してより高性能なシステムを構築することが可能である。

【0154】更に、ホスト計算機とゲスト計算機とで異なるアーキテクチャとして構成することができるため、低コストの入出力装置等を使うことのできるアーキテクチャのシステムをホスト計算機として選択することで、より低コストの複合計算機システムを提供することが可能となる。

【0155】また、入出力装置を識別するための入出力識別テーブルを設けたので、ゲスト計算機は、入出力識別テーブルの識別情報等に基づいて転送相手を識別し特定することが可能となる。従って、ゲスト計算機専用入出力装置と、ホスト計算機に接続されているI/O装置とを併用することができるようになる。従って、ゲスト計算機においてアクセス頻度の高いファイル等をゲスト計算機専用の入出力装置に設けることで、ホストプロセッサにかかる負荷を軽減し、システム全体として高性能

な複合計算機システムを提供することが可能となる。

【0156】また、ゲスト計算機にエミュレーション手段を設けたので、ゲストプロセッサ上で実行される入出力命令を実際に実行することができ、ゲストプロセッサの機械語命令セット及びハードウェアの変更が不要となる。

【0157】また、実行ユニットは、命令デコーダから供給される実行指示内容に基づいて機械語命令の実行を行うことで、エミュレーションを実行させたり、入出力処理あるいは割り込み発生時には割り込み処理を実行させたりすることができる。

【0158】また、エミュレーション手段からの割り込み処理要求等のホスト計算機に対する割り込みの主要因を保持するゲストステータスレジスタを設けたので、ホストプロセッサは、ホスト計算機で割り込みが発生したことを即座に知ることができるとともにその割り込みの主要因を知ることができる。

【0159】また、ゲスト計算機からホスト計算機に対して行う割り込み処理要求を保持する割り込み要求レジスタを設けたので、ホストプロセッサは、ホスト計算機で割り込みが発生したことを即座に知ることができる。

【0160】また、入出力マップテーブルを設けたので、ゲスト計算機制御手段は、入出力処理においてホスト計算機とゲスト計算機との間で転送されるデータの転送元と転送先を特定することが可能となる。

【0161】また、ゲスト計算機制御手段は、ゲスト計算機が管理する仮想的な固定ディスク装置の物理位置とホストOSによって管理される1つのファイルとを対応付けるようにしたので、入出力装置制御手段は、ホスト計算機が提供するファイルシステムをそのまま利用することができる。

【0162】また、ホスト計算機及びゲスト計算機が取り扱うデータの並びの変換を行うバイトオーダ変換機構を設けたので、各計算機においてたとえ異なる形式のデータを取り扱っていたとしてもホスト計算機とゲスト計算機との間の双方向のデータ転送を行うことが可能となる。

【0163】また、ゲストプロセッサに対するホスト計算機からの割り込み要求を受け取る割り込み受信レジスタを設けたので、その割り込み要求の有無をゲストプロセッサに知らせることができる。特に、割り込み受信レジスタにホスト計算機からの入出力実行終了による割り込みであるか、あるいは操作員の指示による割り込みであるかを判別できるようなビット情報を保持できるようにしたので、その割り込み要求元をゲストプロセッサに知らせることができる。

【0164】また、操作員エミュレーション手段を設けたので、ホスト計算機に接続された端末からアドレス情報を設定することなどによりゲスト計算機のゲスト主記憶手段やゲスト制御手段の設定内容を参照することが可

能となる。これにより、ホスト計算機に接続された端末をゲスト計算機と共用することができるので、ゲスト計算機側に別途端末を接続する必要がない。従って、システムを低コストで実現することができる。

【0165】また、本発明に係るインストール/アンインストール方法によれば、第2の計算機で動作するソフトウェアが第1の計算機のファイルとして認識される第1のヘッダ情報と第2の計算機のファイルとして認識される第2のヘッダ情報とを有するように構成したので、単一のインストーラおよび単一のインストール媒体によりインストールでき、インストール作業を効率的に行うことができる。

【0166】また、ホスト計算機とゲスト計算機とにより構成される複合計算機システムにおいて、システム情報を参照して対象ファイルを削除するように構成したので、インストール/アンインストール作業を効率的に行うことができる。

【0167】さらに、インストーラ自身によりソフトウェアの機能を停止した後にインストールを行うように構成したので、インストール作業を迅速に行うことができる。また、アンインストーラ自身によりソフトウェアの機能を停止した後にアンインストールを行うように構成したので、アンインストール作業を迅速に行うことができる。

【図面の簡単な説明】

【図1】 本発明に係る複合計算機システムの一実施例を示す概略的なブロック構成図である。

【図2】 本実施の形態における各計算機で実行されるソフトウェア構成の一実施例を示した図である。

【図3】 本実施の形態における複合計算機システムのハードウェア、ソフトウェア構成例の動作を説明するシステム構成図である。

【図4】 本実施の形態における計算機間通信記憶領域の内部構成例を示した図である。

【図5】 本実施の形態における入出力識別テーブルの構成例を示した図である。

【図6】 本実施の形態におけるゲストステータスレジスタの内部構成例を示した図である。

【図7】 本実施の形態における割り込み受信レジスタの内部構成例を示した図である。

【図8】 本実施の形態における入出力マップテーブルの構成例を示した図である。

【図9】 本実施の形態におけるバイトオーダ変換機構の構成例を示した図である。

【図10】 本実施の形態におけるゲスト計算機において実行される入出力命令において指定された固定ディスク装置に対する入出力動作をホスト計算機上のファイルにマッピングするときの動作を示した図である。

【図11】 本実施の形態におけるゲスト計算機による入出力命令を実行するときの内部処理フローを示した図

である。

【図12】 本実施の形態におけるゲスト 計算機による入出力命令を実行するときの内部処理フローを示した図であり、図11の続きの図である。

【図13】 本実施の形態におけるゲスト 計算機による入出力処理の完了時において発生する入出力終了割り込み処理を実行する際の内部処理フローを示した図である。

【図14】 本実施の形態におけるゲスト 計算機による入出力処理の完了時において発生する入出力終了割り込み処理を実行する際の内部処理フローを示した図であり、
図13の続きの図である。

【図15】 本実施の形態において、操作員とゲスト 計算機とのインタフェース処理を行うときの内部処理フローを示した図である。

【図16】 本実施の形態において、操作員とゲスト 計算機とのインタフェース処理を行うときの内部処理フローを示した図であり、図15の続きの図である。

【図17】 本実施の形態におけるインストール及びアンインストール作業を示す模式図である。

【図18】 本実施の形態において使用するシステム情報の内容例を示した図である。

【図19】 本実施の形態において使用するシステム情報の内容例を示した図である。

【図20】 本実施の形態におけるゲスト 計算機のインストールに用いるファイルの内容を示した図である。

【図21】 本実施の形態におけるインストーラ／アンインストーラの通信経路図である。

【図22】 本実施の形態におけるインストーラ／アンインストーラのシステム停止処理を示したフローチャートである。

【図23】 従来の入出力装置アクセス制御方式の構成例を示した図である。

【図24】 従来例のクライアント・サーバシステムを示す構成図である。

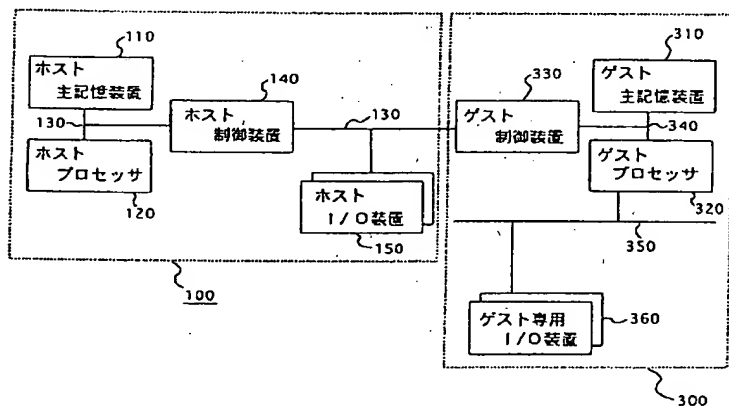
【図25】 従来例のクライアント・サーバシステムにおけるインストール方法及びアンインストール方法を示す模式図である。

【図26】 従来例のインストール方法を説明するために用いる構成図である。

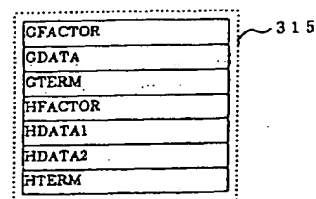
【符号の説明】

100 ホスト 計算機、110 ホスト 主記憶装置、111 ホストオペレーティングシステム(OS)、112 ゲスト 計算機ドライバ、113 I/O装置制御プログラム、114 操作員インタフェース制御プログラム、115 アプリケーションプログラム、116 入出力マップテーブル、120 ホストプロセッサ、130 システムバス、140 ホスト制御装置、150 ホストI/O装置、151 固定ディスク装置、152 磁気テープ装置、153 ローカルエリアネットワーク制御装置、154 操作員I/F装置(端末)、200 インストール媒体、210 ファイル、211 インストーラ、212 アンインストーラ、213 ホストOS用ヘッダ情報、214 インストーラ用のヘッダ情報、300 ゲスト 計算機、310 ゲスト 主記憶装置、311 ゲストオペレーティングシステム(OS)、312 エミュレーションプログラム、313 アプリケーションプログラム、314 チャネルプログラム、315 計算機間通信記憶領域、320 ゲストプロセッサ、321 命令デコーダ、322 実行ユニット、323 入出力識別テーブル、330 ゲスト制御装置、331 システムバスI/F制御部、332 バイトオーダ変換機構、333 割り込み要求レジスタ、334 ゲストステータスレジスタ、335 割り込み受信レジスタ、336、337 マルチプレクサ、338、339 バッファ、340 ゲスト主記憶バス、350 I/Oバス、360 ゲスト専用入出力(I/O)装置。

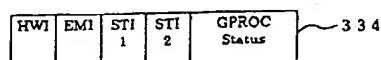
【図1】



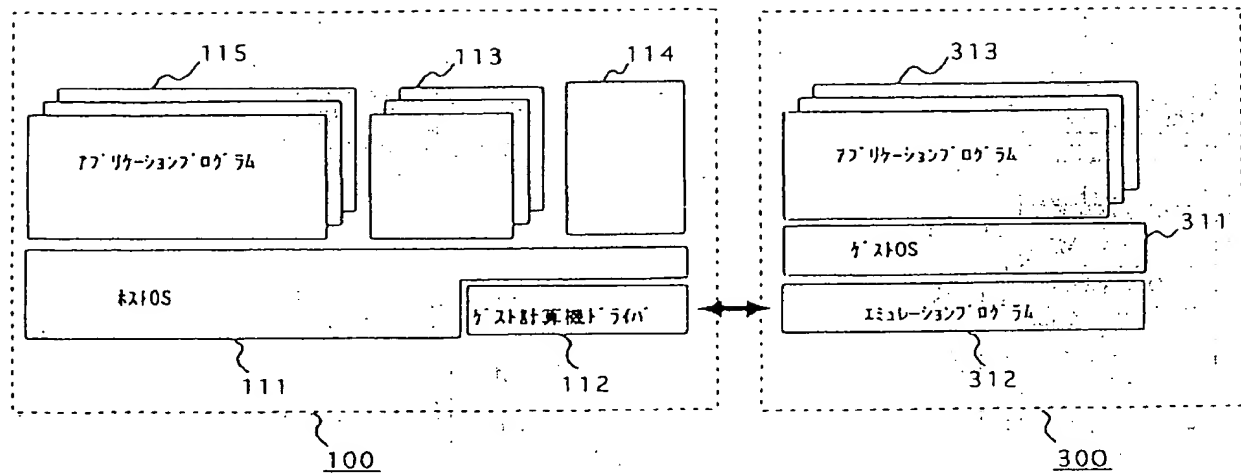
【図4】



【図6】



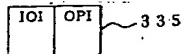
【 図2 】



【 図5 】

I/O装置アドレス	対象となる入出力装置
IOA=x'10	ホストI/O装置
IOA=x'11	ホストI/O装置
IOA=x'90	ゲスト専用I/O装置

【 図7 】

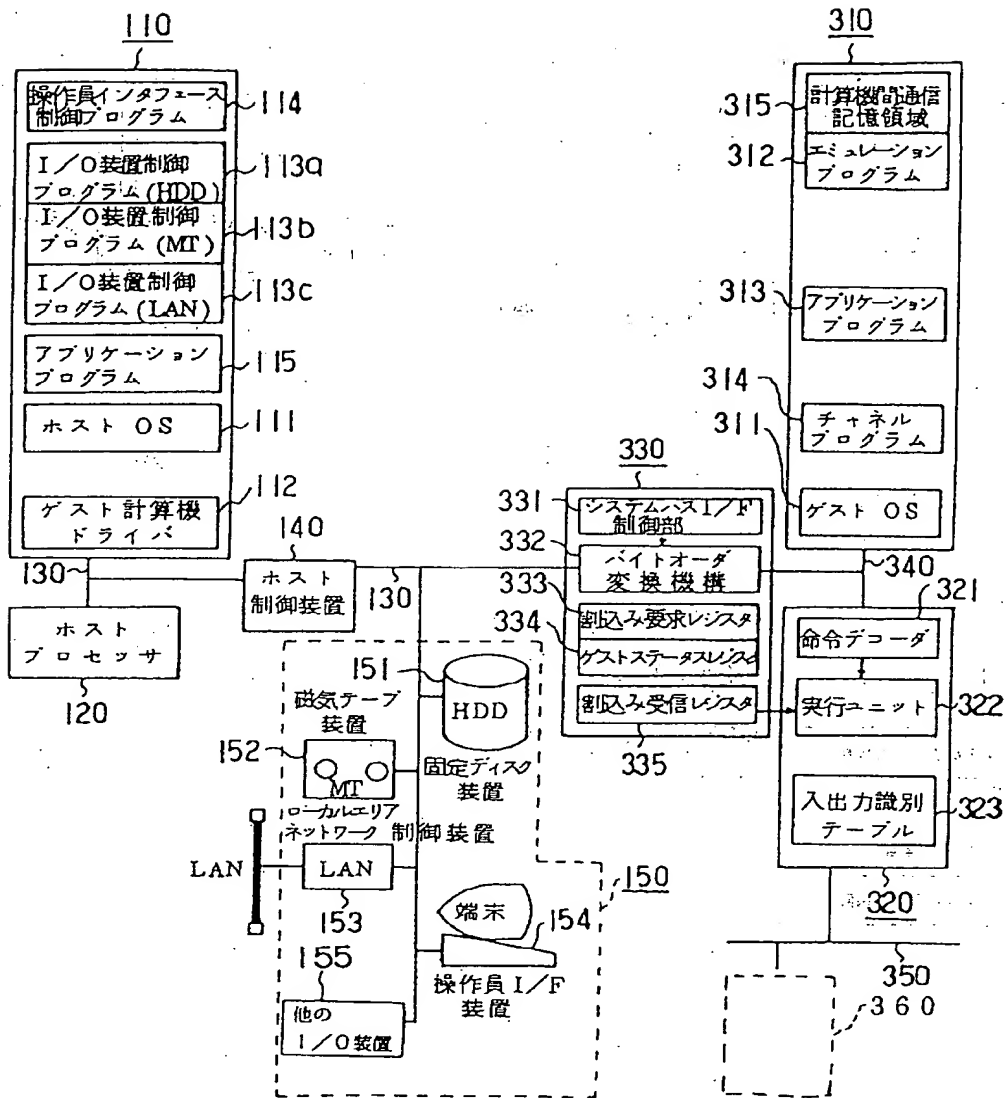


【 図8 】

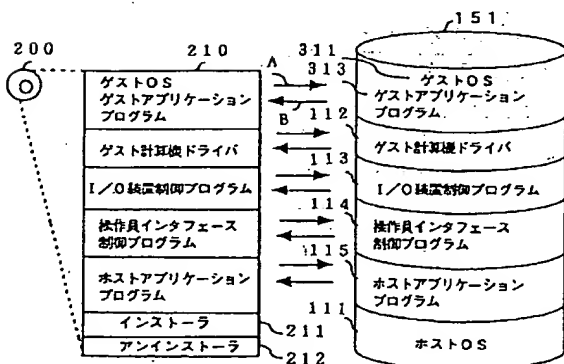
ゲスト計算機でのI/O装置アドレス	ゲスト計算機のI/O装置種類	ホストI/O装置、或はファイル名
IOA=x'10	固定ディスク装置	D:¥GIO¥HDD01
IOA=x'11	固定ディスク装置	E:¥GIO¥HDD02
IOA=x'20	磁気テープ装置	MT-SCSI1-1
IOA=x'30	ローカルネットワーク制御装置	LAN1
:		:

116

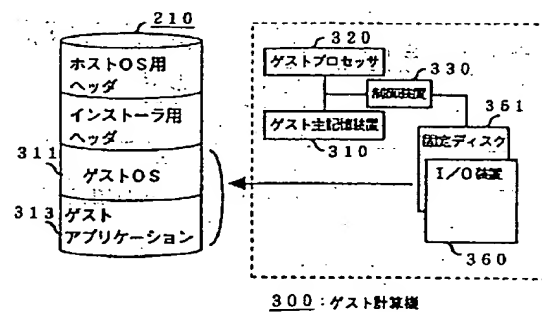
【 図3 】



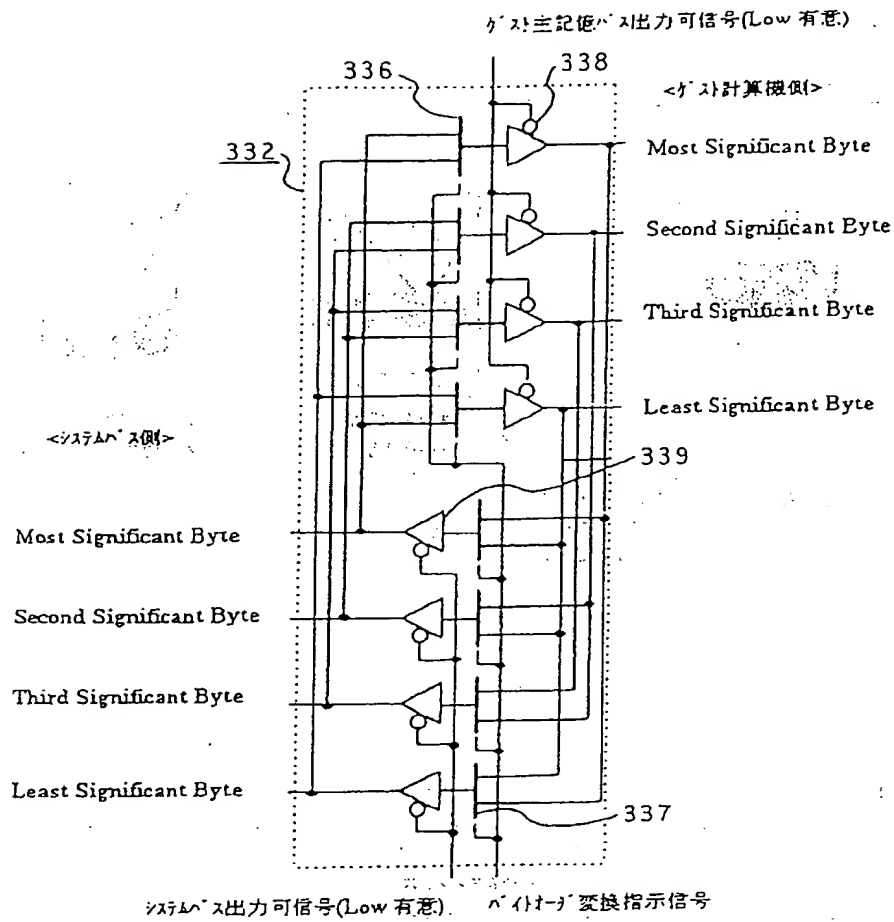
【 図17 】



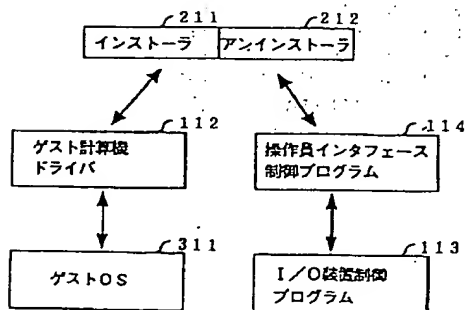
【 図20 】



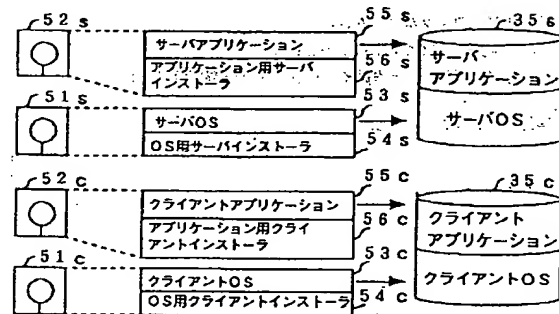
【 図 9 】



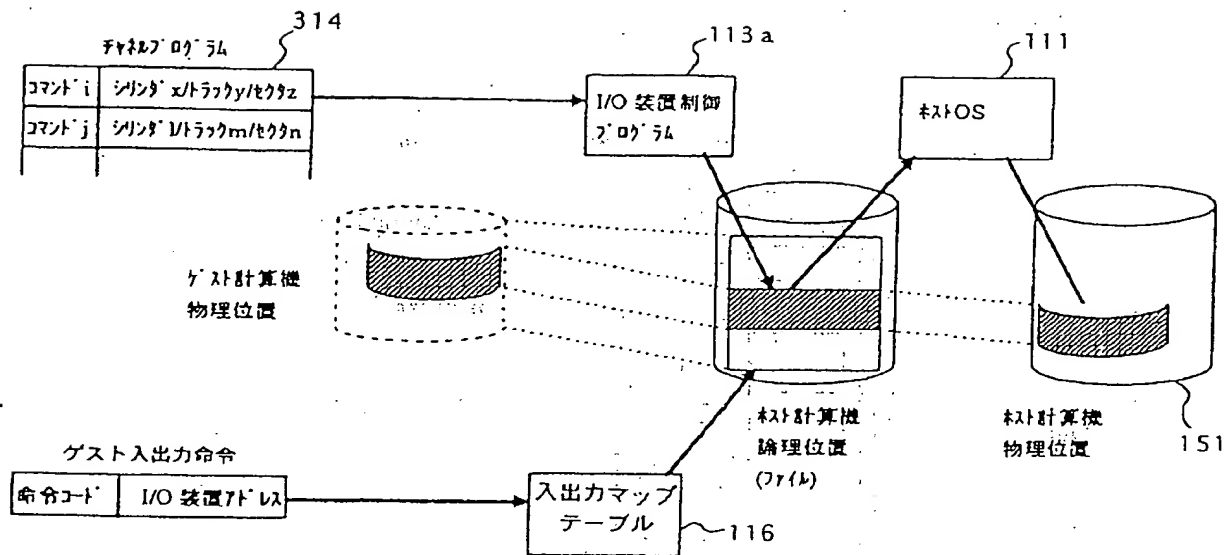
【 図 2 1 】



【 図 2 5 】



【 図10 】



【 図18 】

ゲスト計算機制御プログラム

- ゲスト計算機ドライバ
 - バス名 C:\%GCTRL%\GSTDRV
 - 依存プログラム なし
- 操作員インタフェース制御プログラム
 - バス名 C:\%GCTRL%\IFCTRL
 - 依存プログラム ゲスト計算機ドライバ
- I/O装置制御プログラム (HDD)
 - バス名 C:\%GCTRL%\HDDCTRL
 - 依存プログラム 操作員インタフェース制御プログラム
- I/O装置制御プログラム (MT)
 - バス名 C:\%GCTRL%\MTCTRL
 - 依存プログラム 操作員インタフェース制御プログラム
- I/O装置制御プログラム (LAN)
 - バス名 C:\%GCTRL%\LANCTRL
 - 依存プログラム 操作員インタフェース制御プログラム
- I/O装置制御プログラム (...)

【 図19 】

I/O装置

- 固定ディスク装置
 - IOA=x'10
 - ドライブ D:
 - ファイル名 D:\%GIO%\HDD01
 - サイズ 512Mバイト
 - IOA=x'11
 - ドライブ E:
 - ファイル名 E:\%GIO%\HDD02
 - サイズ 600Mバイト
- 磁気テープ装置
 - IOA=x'20
 - SCSI PortID 0x0
 - SCSI BusID 0x0
 - SCSI TargetID 0x3
 - SCSI Logical Unit ID 0x0
- ローカルエリアネットワーク制御装置
 - IOA=x'30
 - カードID 1

【 図 1 1 】

[illegible]

【 図12 】

ホストプログラム	ホスト計算機H/W	システム	ゲスト計算機H/W	ゲスト計算機プログラム
ホストプログラムの起動開始				ホスト計算機プログラムがI/O命令で指定されたI/O装置(GFACIOR内)をI/O装置として状態を呼び出し
I/O装置制御プログラムの起動開始				I/O動作が可能ならば対応するI/O装置制御プログラムを呼び出し
ゲストプログラムの格納先頭アドレスよりアドレスを1つ読み出し				I/O命令実行終了プログラム、及び実行結果をGTERMにセットして戻込み処理終了
I/O装置制御によりマッピングされたホストI/O装置に対してゲストプログラムで指定された動作を解釈しながら実行開始				
ゲストプログラムの逐次実行				
				ゲスト計算機プログラム終了
				ホスト計算機プログラム終了

【 図13 】

ホストプロセッサ	ホスト計算機トラバ	ホスト計算機H/W	システムバス	ホスト計算機H/W	エミレーションプログラム
ホストプロセッサのI/O装置制御プログラムによるホストプログラムの実行					
ホストプログラムの実行中での入出力処理終了後					
I/O終了割込み要求をホストに受渡す					
I/O終了割込みのホスト計算機による受け付け待ち					
	ホスト計算機からのI/O終了割込み要求を内部へ受け渡す				
	割込み受信レジスタへの割込み要求				
	(もし他のI/O装置制御プログラムからI/O終了割込み要求があった場合には要求を内部へ受け渡す)				
					I/O終了割込みレディ実行開始
					I/O終了割込み受けの割込み要因をGEFACTORにセット
					エミレーションによる割込み要求をホスト計算機に渡す

【 図14 】

ホスト・PPU・リカーション	ゲス計算機・ラウガ	ホスト計算機 H/W	システムバス	ゲス計算機H/W	ミューレ・ラウガ・ラウガ
	ゲス計算機割込み処理 (EMI)開始	ホスト・PPUに 割込み発生	↓	割込み要求レジスタに セット	1/0終了割込みのホスト 計算機による処理は 了済み(GTERMの終了 まで)
	GFACTORにセットされた内 容から1/0終了割込み の受け付けを認識				
	ベンディングされている 1/0終了割込み要求の 内、下番優先順位の高 いもの、次の1/0装置割 込みのPPUに対して割 込み受け付けを通知				
1/0終了割込みの受け付けを 認識	受け付けた1/0終了割込 みに対する内部ベン ディング状態を解除				
1/0終了割込みに必要な終 了・リターン等をゲス主記憶に 格納	もじ別の1/0終了割込 み要求が内部ベン ディングされている割込 み受信レジスタの101を再 度セット				
1/0終了割込みのホスト割込 みの終了をGTERMにセット					ホスト計算機による1/0 終了割込み処理の完 了を検出 GTERMをクリア
1/0終了割込み処理を完了					PPU・ラウガ割込みを割込 み処理レジスタにセット
				1/0終了割込みミュー レの機械命令を実行	ミューレ・ラウガ・ラウガ 行終了

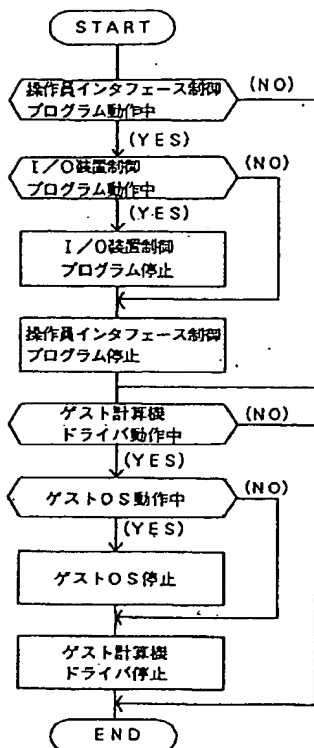
【 図 1 5 】

オペレーション	ゲスト計算機ドライバ	ホスト計算機H/W	ゲストH/W	ゲスト計算機H/W	コメント
操作員I/Fドライバが 操作員からの指示を 受け取り					
ゲスト計算機への処理 内容をH/FACIORにより					
ゲスト計算機での操作 員I/F処理に必要な 付属情報をHDATA1、 HDATA2により					
操作員I/F処理要求 をゲスト受値レジスタ へレジ要求					
ゲスト計算機による処 理終了待ち (INTERの 終了フラグ)	ゲスト受値レジスタ (OP1)のビット		↑	ゲスト受値レジスタ (OP1)のビット	
				ゲストオペタへのゲスト み要求	
				ゲストオペタでのゲスト み受付け、及びゲスト み受値レジスタ(OP1)の み要求	

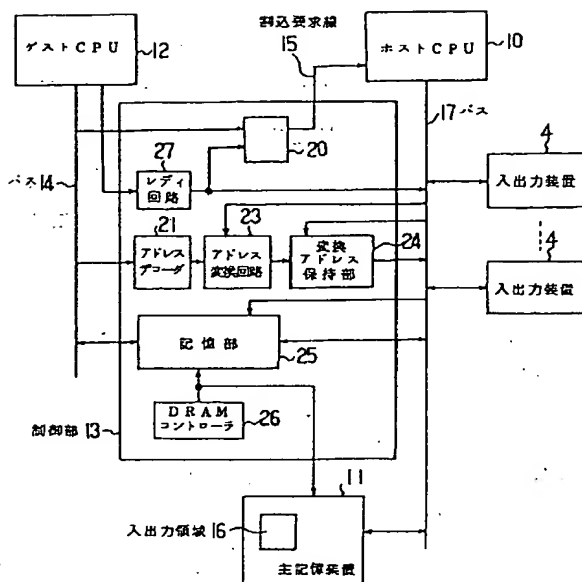
【 図16 】

ホストコンピュータ	ゲスト計算機プログラム	ホスト計算機I/O	ゲスト計算機I/O	ホスト計算機H/W	ゲスト計算機H/W	ミレ-3377のROM
					ミレ-3377のROMの起動	ミレ-3377のROMの起動
						ミレ-3377のROMの起動
						ミレ-3377のROMの起動
ゲスト計算機の処理終了を送出	↓					ミレ-3377のROMの起動
HTERMの内容をミレ-3377のROMに転送						ミレ-3377のROMの起動
ミレ-3377のROMからホスト計算機の使用						ミレ-3377のROMの起動
中のミレ-3377のROMに出力						ミレ-3377のROMの起動
ミレ-3377のROMからホスト計算機の使用						ミレ-3377のROMの起動
ミレ-3377のROMからホスト計算機の使用						ミレ-3377のROMの起動

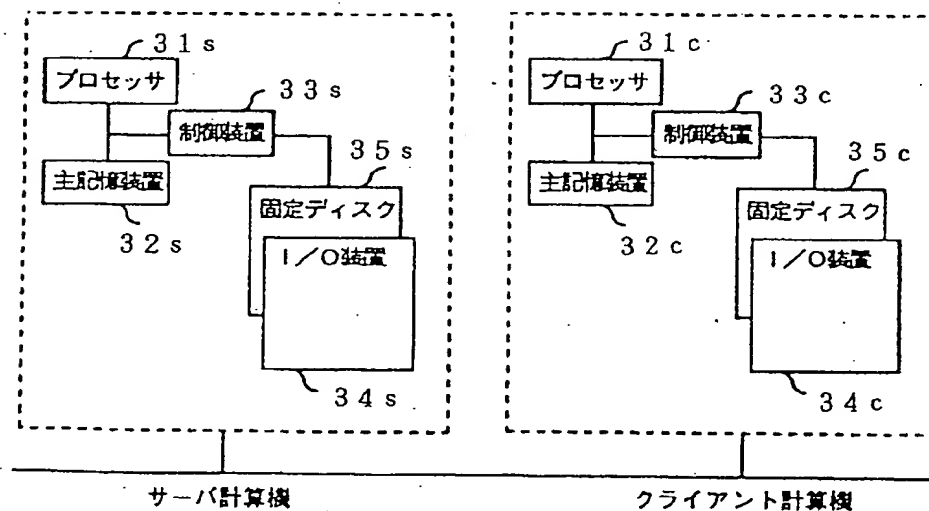
【 図2 2 】



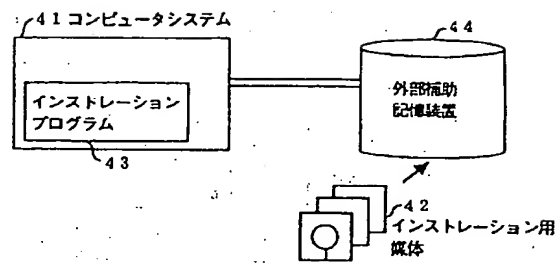
【 図 2 3 】



【 図24 】



【 図26 】



フロント ページの続き

(72)発明者 藤原 聡子

東京都千代田区丸の内二丁目2 番3 号 三
菱電機株式会社内